IJNS
International Journal of Network Security

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

Vol. 24, No. 1 (January 2022)

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: 24, No: 1 (January 1, 2022)

International Journal of Network Security

Information Entropy Models and Privacy Metrics Methods for Privacy Protection

Hui Xia[1] and Weiji Yang[2]
(Corresponding author: Weiji Yang)
School of Software, Shenyang Normal University[1]
School of Life Science, Zhejiang Chinese Medical University[2]
548 Binwen Rd, Binjiang, Hangzhou, Zhejiang, China
Email: yangweiji@163.com
(Received Sept. 3, 2019; Revised and Accepted June 6, 2020; First Online Nov. 9, 2021)

Abstract

The quantification of privacy plays an important role in privacy protection. It can be used to solve privacy metrics as a quantitative measure of information. To realize the privacy metrics, some models of privacy information entropy are proposed according to Shannon's Information Theory. Those models include the basic information entropy model of privacy protection, the information entropy model of privacy protection with adversary, the information entropy model of privacy protection with subjective feelings and multi-source information entropy model of privacy protection. In those models, the information owner is assumed to be the sender, privacy attacker is assumed as to be the recipient, and the privacy disclosure course can be regarded as a communication channel. Based on those assumptions, the entropy, mutual information, conditional entropy, and conditional mutual information are introduced to describe measurement of privacy, privacy disclosure, and privacy and disclosure with background knowledge for the privacy protection system. Furthermore, the quantitative evaluation of privacy protection strength and adversary ability is provided to support quantitative risk assessment for privacy disclosure. Finally, the specific information entropy model, measurement and analysis of privacy protection algorithms, and adversary ability are supplied for location privacy protection application. The proposed models and the privacy metrics can be used to analyze and evaluate the privacy protection technology and privacy disclosure risk assessment.

Keywords: Communication Model; Information Entropy Model; Privacy Measurement; Privacy Protection

1 Introduction

The study of privacy protection started earlier, but in recent years, the industry and academia have suddenly attracted much attention because of big data Privacy algorithms are mainly focused on anonymity methods, including K-anonymity, diversity anonymity and t-close anonymity and their derived methods. The privacy metrics originate from the related anonymity algorithm [15]. In the research of anonymity privacy protection algorithm, some scholars pay more attention to the problem of privacy quantification. Especially in the area of location service, location and trajectory anonymity algorithms have a lot of preliminary research on privacy measurement [13, 14]. However, there are many factors involved in privacy disclosure. Designing effective privacy protection algorithms is still a challenging problem. From these analyses, the study of privacy metrics has very important theoretical significance and application value.

Information entropy, as an effective tool for information measurement, has shown its important contribution in the field of communication [2]. Privacy as a certain of information, naturally can be represented as entropy. For this reason, many scholars have some researches on entropy, such as event entropy, anonymous set entropy, conditional entropy and so on [1, 7, 11]. But these researches are more fragmented or focus on for a particular area but not for a unified model, such as location privacy protection. Moreover, its scope of application is also limited. People may have different opinions on the same privacy in our space-time nature. Based on the above analysis, This paper aims to propose the communication framework of Shannon's Information theory [12]. Several privacy protection information entropy models are proposed, Include basic information entropy model, adversary attack model, subjective and privacy protection model with multiple privacy sources. In these models, the information owner is assumed to be the sender, the privacy purveyor is assumed to be the receiver, and the privacy leak channel is assumed to be a communication channel. Based on these assumptions, Privacy information entropy, average mutual information, conditional entropy and conditional mutual information are introduced to describe privacy measure, privacy disclosure measure,

privacy measure and disclosure measure of privacy information system. Based on this, the paper further puts forward the quantitative evaluation of the strength of privacy protection method and the ability of adversary attack, and tries to provide a theoretical support for quantitative risk assessment of privacy disclosure.

Section 2 of this paper describes the related work. Based on the communication model of information entropy, the information entropy model of privacy metric with common characteristics is proposed in Section 3. Section 4 presents a privacy metric and evaluation system based on the model proposed in Section 3. Section 5 applies the privacy measurement method and evaluation system proposed in this paper to prove the privacy protection method effectiveness. Section 6 gives conclusion.

2 Related Work

The information entropy theory proposed by Shannon [12] solves the theoretical basis of information quantification and communication. Earlier information entropy measure considering privacy research is Diaz et al and Serjantov et al, they proposed using information entropy to measure the anonymity of anonymous communication systems. Assuming the attacker's intention is to determine the true identity of the sender (or recipient) of the message, each user in the system is guessed as the true sender or receiver of the message with a certain probability, and the attacker guesses that a user is a real sender or receiver as a random variable X, and uses information entropy $H(X) = -\sum p(x)\log p(x)$ to quantify the privacy level of the system.

Subsequently, many scholars have applied information entropy to some specific areas of privacy metrics, such as location services, social networks and data mining and other fields [3–6, 8–10, 13, 14]. For different schemes, the probability expression of the random variable is different from the way to deal with entropy. In the field of location-based services, in 2007, Hoh et al. [5,6] proposed privacy measurement based on information entropy to measure uncertainty of trajectory tracking, where the probability of a random variable is represented by the probability that each location instance contains the current tracked vehicle trajectory. In 2009, Ma et al. [10] proposed the privacy measurement method of information entropy in the V2X car network system. Among them, the probability of a random variable performance associated with each location information to the probability that a particular user. The method also takes into account the situation that the probability of a random variable is updated over time, the attacker's cumulative information on the impact of system privacy. In the same year, Lin Xin and others [8] for the LBS in the continuous query problem, a continuous query attack algorithm. They point out that the anonymity of the set is no longer suitable as a measure of the anonymity of the algorithm and presents a measure of entropy based on information. Where the probability of a random variable is represented by the probability that each user u_i is the true originator of the query q, the information entropy is calculated as $H(q)$, and the privacy level of the system is measured by $AD(q) = 2H(q)$. In 2011, Shokri et al. divided the metrics of location privacy into accuracy, certainty and correctness: The accuracy measure is the confidence interval of the attacker's guessing event. The deterministic measure is the uncertainty of the attacker's guess. The correctness measure is the probability of the attacker's error. Among them, the accuracy of the measurement is based on the measurement method of information entropy. The probability of random variables for each observed event is the probability of real events.

In 2012, Chen and others [3] for LBS query privacy measurement. The probability of a random variable is represented by the probability that the attacker has no background knowledge or background knowledge, and the user u_i is the conditional probability of the true sender of the query q, and use mutual information $I(U|q;<r,t,q>) = H(U|q) - H(U|<r,t,q>)$ to measure the privacy level of the system. In the same year, Wang Caimei et al [14] for the trajectory in LBS privacy protection method Silent Cascade proposed based on the information entropy of privacy measurement method. The probability of a random variable is expressed as the probability of every possible trajectory of a user. The entropy calculation for a particular user is $H(u_i)$, and use the standard entropy $D(u_i) = H(u_i)/H_{max}(u_i)$ to measure the privacy level of the system. In 2014, the literature [4, 9] used the information entropy to measure the LBS system's privacy level.

In summary, at present, the theoretical system of privacy measurement based on information entropy is fragmented and lacks a unified model foundation. Regarding the issue above, in this paper, we try to regard the privacy protection system as a communication propagation model, and try to discuss the more common privacy measure information entropy model, and solve some basic concepts and basic system of privacy measurement.

3 Privacy Protection Information Entropy Model

The starting point of this paper is to assume that the owner of the information as the sender, privacy seekers (adversaries) is assumed to be the receiver, the privacy of the leakage channel is assumed to be communication channels.

The information set owned by the sender is called the privacy source, represented by the random variable X, X is a privacy message space made up of privacy messages of all discrete basic disclosure events, which is $\{x_1, x_2, \cdots, x_n\}$, where $x_i (i = 1, 2, \cdots, n)$ is the privacy message of the basic disclosure event; The information collection that the receiver gets is called the privacy sink, with a random variable Y said, which is made up of all the

basic privacy messages acquired by the adversary, which is $\{y_1, y_2, \cdots, y_m\}$, where $y_j (j = 1, 2, \cdots, m)$ is a private information obtained by the adversary. Correspondingly, a specific privacy protection algorithm can be seen as a privacy message conversion, encoding method. It can interfere with the privacy message, and then realize the protection of privacy information. Among them, the privacy protection algorithm of the overall structure of privacy protection mechanism space is called the source of privacy protection mechanism. Adversary in a certain background knowledge of the private information mining and analysis of means of privacy attacks, all privacy methods known as privacy attack space.

Based on this assumption, in this section, we propose several privacy information entropy models based on Shannon information theory [12], including privacy protection basic information entropy model, privacy protection information entropy model with adversary attack, information entropy model with subjective perception and privacy protection information entropy model with multiple privacy sources.

3.1 Privacy Protection Basic Information Entropy Model

Here, we first assume that the adversary has no privacy attack ability, the adversary only observes the privacy information through the channel, and only considers the situation of the discrete single privacy source. The model definition is shown in Figure 1.

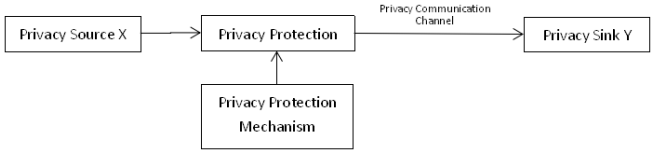


Figure 1: Communication model of privacy protection with single privacy information source

The mathematical model of a single private source X can be expressed as:

$$\begin{bmatrix} X \\ P(X) \end{bmatrix} = \begin{bmatrix} x_1 & x_2 & \cdots & x_i & \cdots & x_n \\ p(x_1) & p(x_2) & \cdots & p(x_i) & \cdots & p(x_n) \end{bmatrix}$$

Among them, $0 \leq p(x_i) \leq 1$, $\sum_{i=1}^{n} p(x_i) = 1$. Similarly, the mathematical model of privacy sink Y can be expressed as :

$$\begin{bmatrix} Y \\ P(Y) \end{bmatrix} = \begin{bmatrix} y_1 & y_2 & \cdots & y_j & \cdots & y_m \\ p(y_1) & p(y_2) & \cdots & p(y_j) & \cdots & p(y_m) \end{bmatrix} \quad (1)$$

Among them, $0 \leq p(y_j) \leq 1$, $\sum_{j=1}^{m} p(y_j) = 1$. For this model, we define the entropy $H(X)$:

$$H(X) = -\sum^{n} p(x_i) \log_2 p(x_i)$$

$H(X)$ is used to characterize the average amount of privacy information of privacy sources, and also the degree of privacy uncertainty of privacy sources. The greater the $H(X)$, the less likely the privacy disclosure is, so it can also be used to measure the degree of privacy protection. In the absence of external conditions, the value is a certain value. When the privacy of the client Y to obtain some privacy information in the conditions, as to the degree of uncertainty of privacy sources, we can introduce the privacy condition entropy $H(X|Y)$, which is defined as

$$H(X|Y) = -\sum_{j=1}^{m} \sum_{i=1}^{n} p(x_i y_j) \log_2 p(x_i|y_j).$$

The conditional entropy means that the privacy source X is still uncertain after receiving the Y information. The degree of uncertainty is caused by the interference (privacy protection) of the privacy leak channel, which is adversaries in the long-term observation of the source of privacy in the process, due to the protection of privacy protection mechanism, the opponent of the source of privacy is still some unknown and easy to prove. This entropy of privacy information satisfies the basic properties of Shannon source entropy [2]. That has non-negative, symmetry, expansibility, certainty, additivity, extreme value, convexity and so on, and satisfies the maximal discrete entropy theorem, will not repeat here. The following describes the average privacy mutual information $I(X;Y)$ to describe the degree of privacy disclosure on the channel, defined as

$$I(X;Y) = -\sum_{i=1}^{n} \sum_{j=1}^{m} p(x_i y_j) \log_2 \frac{p(x_i|y_j)}{p(x_i)} \quad (2)$$

$I(X;Y)$ denotes the average amount of information exchanged between the privacy source X and the privacy sink Y, that is, the amount of privacy information propagated on the channel. It can just depict the overall degree of disclosure of privacy, which can be used as a measure of privacy disclosure.

3.2 Information Entropy Model of Privacy Protection with Adversary Attack

The privacy entropy model proposed in the previous section describes objectively the privacy measurement problem in the case of invulnerability or non-attack capability. In the actual system, there is often a privacy attack analysis, the adversary can be in a certain background knowledge of attack analysis, the model definition shown in Figure 2.

In this model, Z represents the background knowledge

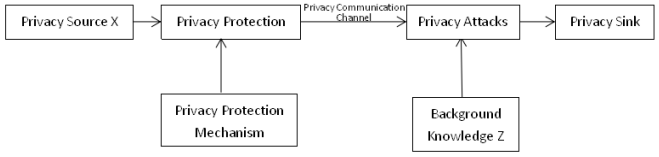


Figure 2: Communication model of privacy protection with single privacy information source and adversaries' attacks

space, and its mathematical model can also be defined as

$$\begin{bmatrix} Z \\ P(Z) \end{bmatrix} = \begin{bmatrix} z_1 & z_2 & \cdots & z_k & \cdots & z_l \\ p(z_1) & p(z_2) & \cdots & p(z_k) & \cdots & p(z_l) \end{bmatrix},$$
$$0 \le p(z_k) \le 1, \sum_{k=1}^{l} p(z_k) = 1, k = 1, 2, \cdots, l$$

The attacker can exploit the background knowledge Z to strengthen the privacy attack. For the attacker, it can combine the privacy message Y and the background knowledge Z to carry on the privacy analysis attack and introduce the attack condition entropy:

$$H(X|YZ) = -\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{k=1}^{l} p(x_i y_j z_k) \log_2 p(x_i|y_j z_k).$$

$H(X|YZ)$ reflects that the attacker obtains the privacy message Y and the background knowledge Z, with regard to the uncertainty that X still exists. It can be used as an attack in the means of privacy under the uncertainty of information can also be used as a measure of privacy protection strength. Similarly, the average mutual information of privacy attacks is further defined:

$$I(X;Y|Z) = \sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{k=1}^{l} p(x_i y_j z_k) \log_2 \frac{p(x_i z_k|y_j}{p(x_i|z_k)p(x_j|z_k)}$$

$I(X;Y|Z)$ reflects the average mutual information between X and Y under the condition that Z is obtained, that is, the amount of privacy information obtained by the receiver and the degree of privacy disclosure with background knowledge attack.

3.3 Information Entropy Model with Subjective Feeling

In reality, the sensitivity of privacy information is usually subjective, and different people feel different about the value of privacy information. In this section, the weights are introduced into the information entropy model of the first two sections, and the information entropy model with subjective feelings are proposed and measurement.

1) Entropy model of privacy protection with subjective feelings
For the privacy message $x_i (i = 1, 2, \cdots, n)$ of the communication model described in Figure 1, a non-negative real number is set as the sensitivity weight of the message. The greater the weight is, the greater the sensitivity. The weight space is as follow

$$\begin{bmatrix} X \\ W(X) \end{bmatrix} = \begin{bmatrix} x_1 & x_2 & \cdots & x_n \\ p(x_1) & p(x_2) & \cdots & p(x_n) \end{bmatrix},$$
$$w_i \ge 0, i = 1, 2, \cdots, n$$

Define privacy-weighted source entropy:

$$H_w(X) = -\sum_{i=1}^{n} w_i p(x_i) \log_2 p(x_i).$$

$H_w(X)$ is to describe the subjective sensitivity of different users to privacy messages by weight $W_i (i = 1, 2, \cdots, n)$, so as to realize the privacy information measure with subjective feelings. The weighted entropy of private sources obviously has the following properties:

Non-negative: That the source of privacy in the event of a privacy disclosure event, which can always provide some privacy information.

Continuity: When the probability of private event occurs, the private information source will form another privacy source. The weight entropy of the two privacy sources before and after the change is continuous. This feature is very effective in characterizing changes in the characteristics of a source of privacy due to temporal changes. Such as in a certain period of time, a person's life law is fixed, leading to its ability to disclose personal privacy behavior model of the probability distribution is relatively fixed,, but with the passage of time, the person's life will be a continuous pattern of minor changes, and thus be able to reveal their privacy behavior pattern of probability distribution has also been a slight change. However, before and after the behavior change, the weighted entropy of the behavioral population is continuous.

In addition, the weighted entropy of the source information and other properties of information entropy, in the privacy protection system also have the corresponding practical significance.

Likewise, we can define the privacy weighting condition entropy $H_w(X|Y)$ to describe the condition that privacy seekers obtain some privacy information, about the owner's privacy information uncertainty of average:

$$H_w(X|Y) = -\sum_{i=1}^{n}\sum_{j=1}^{m} p(x_i y_j) \log_2 p(x_i|y_j).$$

Define privacy-weighted average mutual information $I_w(X|Y)$ to describe the degree of privacy information disclosure with subjective feelings, under the

protection of the privacy protection mechanism. It indicates the amount of private information obtained by privacy seekers by observing the privacy events:

$$I_w(X;Y) = \sum_{i=1}^{n} w_i \sum_{j=1}^{m} p(x_i y_j) \log_2 \frac{p(x_i|y_j)}{p(x_i)}$$

Here, the privacy-weighted condition entropy and the privacy-weighted average mutual information only take into account the subjective feelings and preferences of the privacy source to the privacy message. In the actual system, not only the information owner of their own privacy information has different subjective feelings, privacy seekers to obtain the privacy information also have different subjective feelings and preferences. It can further explore the private communication model of privacy in the privacy of the subject of privacy messages and give weight to the feelings, and even set up the weight matrix that depicts the preference source of privacy source and privacy sink.

2) An Entropy Model of Privacy Protection with Subjective Feelings and Attacking
In consideration of the subjective feelings or preferences of the privacy owner of his privacy information, we define the weighting attack condition entropy $H_w(X|YZ)$ to describe the attack effect of privacy sink Y in background knowledge Z support, it can also be used as a measure of privacy protection against rival attacks.

$$\begin{aligned} H_w(X|YZ) &= -\sum_{i=1}^{n} w_i \sum_{j=1}^{m} \sum_{k=1}^{l} p(x_i y_j z_k) \\ &\quad \cdot \log_2 p(x_i|y_j z_k). \end{aligned}$$

On this basis, further defined privacy attacks weighted average mutual information $I_w(X;Y|Z)$; It represents the amount of private information received by the privacy sink under the condition that Z is obtained, which characterizes the privacy disclosure measure under the condition of background knowledge:

$$\begin{aligned} I_w(X;Y|Z) &= \sum_{i=1}^{n} w_i \sum_{j=1}^{m} \sum_{k=1}^{l} p(x_i y_j x'_k) \\ &\quad \cdot \log_2 \frac{p(x_i z_k|y_j)}{p(x_i|z_k)(y_j|z_k)}. \end{aligned}$$

3.4 Multi-source Privacy Privacy Protection Information Entropy Model

Realistic information system owners often have more than one. Thus involving a number of privacy sources of the problem. Therefore, it is necessary to establish a privacy-protected communication model with multiple privacy sources to measure the protection and attack of the privacy information of multiple sources associated with each other. Figure 3 shows the privacy of multiple privacy sources without privacy attacks communication model.

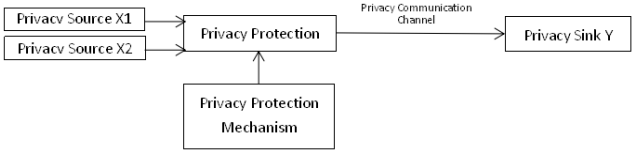


Figure 3: Communication model of privacy protection with multi-source of privacy information and none privacy attacks

In the communication model shown in Figure 3, privacy source X_1 and privacy source X_2 together constitute the source of privacy X. The mathematical model is

$$\begin{bmatrix} X_1 \\ P(X_1) \end{bmatrix} = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1n_1} \\ p(x_{11}) & p(x_{12}) & \cdots & p(x_{1n_1}) \end{bmatrix} \quad (3)$$

$$0 \le p(x_{i_1}) \le 1, \sum_{i_1=1}^{n_1} p(x_{i_1}) = 1, i_1 = 1, 2, \cdots, n_1$$

$$\begin{bmatrix} X_2 \\ P(X_2) \end{bmatrix} = \begin{bmatrix} x_{21} & x_{22} & \cdots & x_{2n_2} \\ p(x_{21}) & p(x_{22}) & \cdots & p(x_{2n_2}) \end{bmatrix} \quad (4)$$

$$0 \le p(x_{i_2}) \le 1, \sum_{i_2=1}^{n_2} p(x_{i_2}) = 1, i_2 = 1, 2, \cdots, n_2$$

The mathematical model of privacy sink Y is described by Equation (1). The definition of multi-source joint source of privacy information source entropy $H(X_1X_2)$. The source entropy characterizes a number of privacy measures with associated privacy owners:

$$\begin{aligned} H(X_1X_2) &= -\sum_{i_1=1}^{n_1} \sum_{i_2=1}^{n_2} p(x_{i_1}x_{i_2}) \log_2 p(x_{i_1}x_{i_2}) \\ &= H(X_1) + H(X_2|X_1). \end{aligned}$$

The multi-source federated privacy condition entropy of private source X under the condition of known privacy client Y can be defined as $H(X|Y) = H(X_1X_2Y) - H(Y)$. The definition of the definition of a number of associated with the source of privacy in the implementation of privacy protection. The average degree of joint uncertainty of the privacy information is obtained by the privacy information acquirer after observing the privacy event.

Simultaneously, the multi-source joint average mutual information $I(X_1X_2;Y)$ can be defined to characterize the degree of privacy disclosure of the associated plurality of privacy sources:

$$I({}_1X_2;Y) = \sum_{i_1=1}^{n_1} \sum_{i_2=1}^{n_2} \sum_{j=1}^{m} p(x_{i_1}x_{i_2}y_j) \log_2 \frac{p(x_{i_1}x_{i_2}|y_j)}{p(x_{i_1}x_{i_2})}$$

- Privacy protection information entropy model with multiple privacy sources and privacy attack

Based on the information entropy model with privacy attack, privacy protection was proposed in Section 2.2, introducing a plurality of associated information owners, forming a new associated multi-privacy source, constructing the privacy protection information entropy model of multi-privacy source with adversary attack, as shown in Figure 4.

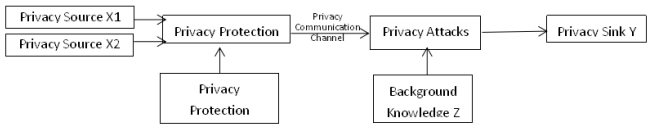


Figure 4: Communication model of privacy protection with multi-source of privacy information and attacks

Figure 4 shows the source model of the communication model as shown in Equation (3) and Equation (4), the mathematical model of privacy sink Y is described in Equation (1). The multi-source joint source entropy of this model is $H(X) = H(X_1X_2)$, multi-source joint privacy attack condition entropy is $H(X_1X_2|YZ)$, multi-source joint privacy attack condition average mutual information is $I(X_1X_2;Y|Z)$, among them, multi-source joint privacy attack condition entropy represents the uncertainty of the privacy information of the joint privacy source under background knowledge attack; multi-source federated privacy attack condition average mutual information is the degree of privacy disclosure of the joint privacy source under background knowledge attack, among them,

$$\begin{aligned} H(X_1X_2|YZ) &= H(X_1X_2|YZ) - H(YZ) \\ I(X_1X_2;Y|Z) &= H(X_1X_2) - H(X_1X_2Y|Z). \end{aligned}$$

4 Privacy Metrics and Their Evaluation Mechanisms

Information entropy and average mutual information can be related to the measurement of privacy information, based on this, the establishment of anti-attack ability evaluation method of privacy protection mechanism is realized.

4.1 Privacy Measurement Methods

In the basic information entropy model, the privacy condition entropy $H(X|Y)$ can be used to measure in the privacy protection mechanism. The privacy source still has a degree of uncertainty so that it can evaluate the strength of the privacy protection algorithm. If the A records for a specific privacy protection, then $H_{p_i}(X|Y)$ is in the implementation of protection with $I(X,Y)$ after the privacy of the destination (adversary) Y is still on the unknown amount of privacy, from a privacy owner's perspective, it is desirable that the condition $I(X,Y)$ indicates that the privacy information X is protected by the privacy protection mechanism. The average amount of privacy information acquired by the sink Y, similarly, $I_{p_i}(X,Y)$ is the privacy information received by Y after being protected by p_i. It should be as small as possible.

Property 1. *Privacy condition entropy $H(X|Y)$ and privacy mutual information $I(X;Y)$ have the consistency of privacy measure.*

Proof. From Equation (2) shows:

$$\begin{aligned} I(X;Y) &= \sum_{i=1}^{n}\sum_{j=1}^{m} p(x_iy_j)\log_2\frac{p(x_i|y_j)}{p(x_i)} \\ &= \sum_{i=1}^{n}\sum_{j=1}^{m} p(x_iy_j)\log_2\frac{1}{p(x_i)} \\ &\quad - \sum_{i=1}^{n}\sum_{j=1}^{m} p(x_iy_j)\log_2\frac{1}{p(x_i|y_j)} \\ &= \sum_{i=1}^{n} p(x_i)\log_2\frac{1}{p(x_i)} \\ &\quad - \sum_{i=1}^{n}\sum_{j=1}^{m} p(x_iy_j)\log_2\frac{1}{p(x_i|y_j)} \\ &= H(X) - H(X|Y). \end{aligned}$$

So there $I(X;Y) = H(X) - H(X|Y)$. The larger the privacy condition entropy, the smaller the mutual information of privacy meaning, and they are consistent. □

4.2 Privacy Protection Mechanism and Privacy Attack Evaluation and Analysis

4.2.1 Privacy protection of the Basic Information Entropy Model

The purpose of the privacy protection mechanism (algorithm) is to protect the privacy of information owners. The aim is to make $H(X|Y)$ as small as possible. That is, through some kind of privacy protection mechanism, so that the amount of information $I(X;Y)$ obtained by the privacy gatherer is as small as possible, preferably 0.

Definition 1. *If under the protection of some kind of privacy protection, privacy, average mutual information $I(X;Y)$ (private sink received from the source to the privacy of the privacy information is 0), it is said that the privacy protection mechanism for this source is completely privacy protection.*

Definition 2. *For the same privacy source X, privacy protection mechanisms p_i and p_j are used to protect:*

- *If $H_{p_i}(X|Y) < H_{p_j}(X|Y)$ or $I_{p_i}(X;Y) > I_{p_j}(X;Y)$, then the privacy protection mechanism p_j is better than the p_i privacy protection. Abbreviated as partial order relation $p_i \prec p_j$.*

- *If $H_{p_i}(X|Y) = H_{p_j}(X|Y)$ or $I_{p_i}(X;Y) = I_{p_j}(X;Y)$, then said the privacy protection mechanism p_i and p_j privacy protection equivalence. Abbreviated as equivalence relation $p_i \cong p_j$.*

Theorem 1. *Partial order relation and equivalence relation of privacy protection mechanism are defined as defined in definition 2, then partial ordering relation is transitivity, and equivalence relation has reflexivity, transitivity and symmetry.*

Proof. If $p_i \prec p_j$, $p_j \prec p_k$, and by definition there are $H_{p_i}(X|Y) < H_{p_j}(X|Y)$ and $H_{p_j}(X|Y) < H_{p_k}(X|Y)$. According to the nature of information entropy, easy to get $H_{p_i}(X|Y) < H_{p_k}(X|Y)$, namely $p_i \prec p_k$. Similarly, if $I_{p_i}(X;Y) > I_{p_j}(X;Y)$ and $I_{p_j}(X;Y) > I_{p_k}(X;Y)$, according to the nature of 1, easy to prove $I_{p_i}(X;Y) > I_{p_k}(X;Y)$. Then there is $p_i \prec p_k$. That is, partial order relations are transitivity. Similarly, it's easy to verify the equivalence of the three characteristics. □

Definition 3. *(privacy protection validity distance). In the basic entropy model of privacy protection, privacy protection mechanism p_i and p_j are used to protect the same privacy source X, if the amount of private information received by privacy sink Y is $I_{p_i}(X;Y)$ and $I_{p_j}(X;Y)$, respectively. The validity distance between these two privacy protection mechanisms is defined as $d_I = |I_{p_i}(X;Y) - I_{p_j}(X;Y)|$.*

In the privacy protection basic information entropy model, the privacy protection validity distance depicts the effectiveness of two different privacy protection mechanisms to protect the same privacy information. Clearly, d_I smaller, the smaller the difference between the two effectiveness of privacy protection algorithms; The greater the d_I, the greater the difference in the effectiveness of the two privacy protection algorithms.

4.3 The Privacy Protection Mechanism and Privacy Attack Evaluation with Adversary Attack

In the actual system, the goal of the privacy protection mechanism is: In the case of various types of privacy attacks against opponents. The privacy information of the information owner is still made available to the privacy seeker as little as possible. That is, through a privacy protection mechanism to resist the opponent in the certain background knowledge of privacy attacks, making privacy seekers get the amount of private information $I(X;Y|Z)$ as small as possible, preferably 0.

Definition 4. *For a privacy protection system with an adversary attack, if $I(X;Y|Z) = 0$, that is, in the opponent in the context of knowledge Z attack, if the privacy protection mechanism can make the owner of the information disclosure of privacy information is 0, then said the privacy system is perfect privacy protection.*

Definition 5. *For a privacy protection system with an adversary attack, if the adversary adopts the privacy attack means A_r to carry on the attack, the system respectively uses the privacy protection mechanism and to carry on the protection:*

- *If $H_{p_i,A_r}(X|YZ) < H_{p_j,A_r}(X|YZ)$ or $I_{p_i,A_r}(X;Y|Z) < I_{p_j,A_r}(X;Y|Z)$, then in the resistance to A_r attack, the privacy protection mechanism p_j is better than the p_i privacy protection, recorded as partial order relation $p_i(A_r) \prec p_j(A_r)$;*
- *If $H_{p_i,A_r}(X|YZ) = H_{p_j,A_r}(X|YZ)$ or $I_{p_i,A_r}(X;Y|Z) = I_{p_j,A_r}(X;Y|Z)$, then it is said that the privacy protection mechanism p_i and p_j privacy protection equivalence, recorded as partial order relation $p_i(A_r) \cong p_j(A_r)$;*

Definition 6. *(Anti-privacy attack privacy protection validity distance). In the privacy entropy model with the adversary attack, the same privacy source X and privacy attack A_r. If under the privacy attack, the use of privacy protection mechanisms p_i and p_j protection respectively, and the amount of private information obtained by privacy sink Y is $I_{p_i,A_r}(X;Y|Z)$ and $I_{p_j,A_r}(X;Y|Z)$. The validity distance between the two privacy protection mechanisms in the privacy attack A_r is $d_I(A_r) = |I_{p_i,A_r}(X;Y|Z) - I_{p_j}(X;Y|Z)|$.*

In the information entropy model of privacy protection with adversary attack, the validity distance $d_I(A_r)$ of the privacy protection anti-privacy attack depicts the validity difference of different privacy protection mechanisms under the same privacy attack.

Definition 7. *Under the same privacy protection mechanism p_i, the adversary adopts privacy attack A_r and A_q to attack:*

- *If $H_{p_i,A_r}(X|YZ) > H_{p_i,A_q}(X|YZ)$ or $I_{p_i,A_r}(X;Y|Z) > I_{p_i,A_q}(X;Y|Z)$, it is said that under the protection of privacy protection mechanism p_i, privacy attack A_r more effective than the A_q, recorded as partial order relation $A_r(p_i) \succ A_a(p_i)$.*
- *If $H_{p_i,A_r}(X|YZ) = H_{p_i,A_q}(X|YZ)$ or $I_{p_i,A_r}(X;Y|Z) = I_{p_i,A_q}(X;Y|Z)$, then it is said that under the protection of privacy protection mechanism p_i, private attack A_r and privacy attacks A_q privacy attack effectiveness equivalent, recorded as partial order relation $A_r(p_i) \cong A_q(p_i)$.*

Theorem 2. *If partial relation and equivalence relation are defined as Definition 5 or Definition 7, partial ordering relation satisfies transitivity, and equivalence relation $\cong$ satisfies reflexivity, symmetry and transitivity.*

Proof. Similar to the proof of Theorem 1, abbreviated. □

Definition 8. *(privacy attack validity distance). In the privacy protection information entropy model with the adversary attack, under the same privacy protection mechanism p_i, adversaries use privacy attacks A_r and A_q to*

attack, The validity distance between the two privacy attacks is called $d_I(p_i) = |I_{p_i,A_r}(X;Y) - I_{p_i,A_q}(X;Y)|$.

In the information entropy model of privacy protection with adversary attack, the validity of privacy attack distance $d_I(p_i)$ depicts the difference of adversary attack ability, which gives the measure of adversary attack ability.

Theorem 3. *In the privacy protection communication model with the adversary attack, suppose the rival's background knowledge is* Z. *Then* $I(X;Y) \leq I(X;YZ)$.

Proof. From the calculation of the average mutual information equation:

$$\begin{aligned} I(X;Y) &= H(X) - H(X|Y) \\ I(X;YZ) &= H(X) - H(X|YZ). \end{aligned}$$

Let (22) be subtracted from (21) to obtain:

$$I(X;YZ) - I(X;Y) = H(X|Y) - H(X|YZ).$$

By the nature of information entropy $H(X|Y) \geq H(X|YZ)$, so $H(X|Y) - H(X|YZ) \geq 0$ and $I(X;YZ) \geq I(X;Y)$. □

The theorem states: adversary in a certain background knowledge of privacy attacks and analysis, the adversary to obtain the privacy information is not less than its background information cannot get the privacy information. This also provides a direction for privacy protection, that is, the privacy information intercepted by the adversary and the background information associated with it as small as possible, so as to maximize the protection of privacy information.

5 Experiments and Simulations

The privacy protection information entropy model proposed above and its measurement method belong to the general situation, applicable to different scenarios. The following is a simple location privacy protection application for the example of the effectiveness of the model analysis. Assume that a user A_u moves in an area divided into M blocks, denote $R = \{r_1, r_2, \cdots, r_M\}$ is the set of different areas of M blocks, that is, the location space, the purpose of the attacker is to determine the user's real location.

5.1 Location Privacy Protection Communication Model

Corresponding to the privacy protection information entropy model with adversary attack, the privacy source is the location distribution R for which the user may be located, and the value of the random variable R indicates that the user u is in a certain location area r_i, using $\{r_1, r_2, \cdots, r_m\}$ indicates the position of the region in which the user space, assuming that the probability of each region is $p(r_i)$, has $0 \leq p(r_i) \leq 1$, $\sum_{i=1}^{M} p(r_i) = 1$, the probability model of R can be expressed as

$$\begin{bmatrix} R \\ P(R) \end{bmatrix} = \begin{bmatrix} r_1 & r_2 & \cdots & r_i & \cdots & r_M \\ p(r_1) & p(r_2) & \cdots & p(r_i) & \cdots & p(r_M) \end{bmatrix}$$

The true location distribution information of the user is privacy information, in order to prevent the attacker from obtaining the real area of the user directly, it is necessary to protect the position distribution R of the user, through a location privacy protection mechanism (Including location generalization, taking pseudonyms, hiding or adding virtual locations, etc.) after performing the privacy protection process on the position distribution R, becomes an observable position distribution R' that can be directly observed by an attacker, set up $R' = \{r'_1, r'_2, \cdots, r'_M\}$, where r'_i is the area of the user u that can be observed by the attacker after privacy protection, and the probability model of observable position distribution R' is

$$\begin{bmatrix} R' \\ P(R') \end{bmatrix} = \begin{bmatrix} r'_1 & r'_2 & \cdots & r'_i & \cdots & r'_M \\ p(r'_1) & p(r'_2) & \cdots & p(r'_i) & \cdots & p(r'_M) \end{bmatrix}$$

$$0 \leq p(r'_i) \leq 1, \sum_{i=1}^{M'} p(r'_i) = 1$$

After the attacker obtains the observable position distribution R', combined with the background knowledge of the user u to attack the location, we get the inferred position $\hat{R}$ of the attacker, set up $\hat{R} = \{\hat{r_1}, \hat{r_2}, \cdots, \hat{r_M}\}$, among them, $\hat{R}$ that the attacker guessed the user U is the real region, its probability model is

$$\begin{bmatrix} \hat{R} \\ P(\hat{R}') \end{bmatrix} = \begin{bmatrix} \hat{r_1} & \hat{r_2} & \cdots & \hat{r_i} & \cdots & \hat{rM} \\ p(\hat{r_1}) & p(\hat{r_2}) & \cdots & p(\hat{r_i}) & \cdots & p(\hat{rM}) \end{bmatrix}$$

$$0 \leq p(\hat{r_1}) \leq 1, \sum_{i=1}^{\hat{M}} p(\hat{r_i}) = 1$$

Figure 5 shows the communication model of the location privacy protection scene, which can be regarded as a concrete example of privacy protection information entropy model with adversary attack.

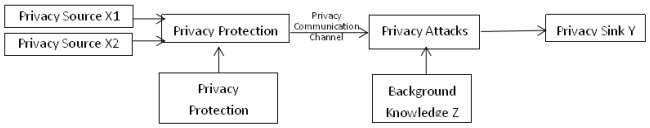


Figure 5: Communication model of location privacy

5.2 Comparison of Different Privacy Protection Mechanisms under the Same

Background Knowledge In the initial stage, the user u is in a real region r_i, The probability that the user is in the

region r_i is 1. The probability of being in other regions is 0. Specifically

$$\begin{bmatrix} R \\ p(R) \end{bmatrix} = \begin{bmatrix} r_1 & r_2 & \cdots & r_i & \cdots & r_M \\ 0 & 0 & \cdots & 1 & \cdots & 0 \end{bmatrix}$$

In this case, the entropy of the entropy of the source information, that is, the location distribution R, is $H(R) = -\sum_{i=1}^{M} p(r_i)\log p(r_i) = 0$.

1) Weak privacy protection intensity of privacy measures
If the location generalization is used as the location privacy protection mechanism, if the user u's publishing location is generalized from area r_i to $\{r_{i-1}, r_i, r_{i+1}, r_{i+2}\}$, the probability model of observable location distribution is as follows:

$$\begin{bmatrix} R' \\ P(R') \end{bmatrix} = \begin{bmatrix} r'_1 & \cdots & r'_{i-1} & r'_i & r'_{i+1} & r'_{i+2} & \cdots & r'_{M'} \\ 0 & \cdots & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \cdots & 0 \end{bmatrix}$$

Then the entropy of observable position distribution is $H(R') = -\sum_{i=1}^{M'} p(r'_i)\log p(r'_i) = 2$, it is equivalent to the entropy containing privacy information entropy model adversary under $H(X|Y)$. After the attacker obtains the observable position distribution, combined with their own background knowledge for analysis, in a certain background knowledge, the probability distribution model of the inferred location of user u is analyzed as follows:

$$\begin{bmatrix} \hat{R} \\ P(\hat{R}) \end{bmatrix} = \begin{bmatrix} \hat{r}_1 & \cdots & \hat{r}_i & \hat{r}_i & \hat{r}_{i+1} & \hat{r}_{i+2} & \cdots & \hat{r}_M \\ 0 & \cdots & \frac{1}{4} & \frac{1}{4} & \frac{1}{8} & \frac{1}{8} & \cdots & 0 \end{bmatrix}$$

At this point we can get $H(\hat{R}) = -\sum_{i=1}^{\hat{M}} p(\hat{r}_i)\log p(\hat{r}_i) = 1.75$, it indicates the degree of uncertainty of the location of the user under the condition of background knowledge, it is equivalent to the entropy privacy information entropy model adversary under $H(X|YZ)$.

2) Strong privacy protection intensity of privacy measures
When we take the generalization of the location area becomes larger, that is, strong privacy protection means, we take the release location of user u from area $\{r_{i-1}, r_i, r_{i+1}, r_{i+2}\}$ to $\{r_i, r_{i+1}, \cdots, r_{i+7}\}$. The probabilistic model of observable location distribution is

$$\begin{bmatrix} \hat{R} \\ P(\hat{R}) \end{bmatrix} = \begin{bmatrix} \hat{r}_1 & \cdots & \hat{r}_i & \hat{r}_{i+1} & \hat{r}_{i+2} & \cdots & \hat{r}_M \\ 0 & \cdots & \frac{1}{2} & \frac{1}{8} & \frac{1}{8} & \cdots & 0 \end{bmatrix}$$

At this point, we can get $H(\hat{R}) = -\sum_{i=1}^{\hat{M}} p(\hat{r}_i)\log p(\hat{r}_i) = 2.3125$, it represents attacker uncertainty measure the user's location under conditions having background knowledge, equivalent to the entropy privacy information entropy model adversary under $H(X|YZ)$. From $2.3125 > 1.75$ can be verified with the adversary attack privacy protection information entropy model $H_{p_i,A_r}(X|YZ) < H_{p_j,A_r}(X|YZ)$ is established.

5.3 Comparison

Comparison of the Effects of Different Privacy Attacks under the Same Privacy Protection Mechanism:

1) The privacy measure of weak privacy attack strength
The privacy measure of the intensity of weak privacy protection is the same as in Section 5.2.

2) Strong privacy attack strength of privacy measures
Privacy protection mechanism with the weak privacy protection in Section 5.2 of the privacy measure, the attacker to obtain the observable position distribution, combined with their background knowledge of the analysis, in strong privacy attack strength. The probability model for the more accurate inferred positional distribution for user u is analyzed as follows:

$$\begin{bmatrix} \hat{R} \\ P(\hat{R}) \end{bmatrix} = \begin{bmatrix} \hat{r_1} & \cdots & \hat{r_i} & \hat{r_i} & \hat{r_{i+1}} & \hat{r_{i+2}} & \cdots & \hat{rM} \\ 0 & \cdots & \frac{1}{6} & \frac{2}{3} & \frac{1}{12} & \frac{1}{12} & \cdots & 0 \end{bmatrix}$$

At this point, we can get $H(\hat{R}) = -\sum_{i=1}^{\hat{M}} p(\hat{r}_i)\log p(\hat{r}_i) = 1.418$, it represents attacker uncertainty measure the user's location under conditions having background knowledge, equivalent to the entropy privacy information entropy model adversary under $H(X|YZ)$. From $1.418 < 1.75$ can be verified with the adversary attack privacy protection information entropy model $H_{p_i,A_r}(X|YZ) < H_{p_i,A_q}(X|YZ)$ is established.

6 Conclusion

In this paper, several privacy protection information entropy models are proposed based on Shannon information theory. The key point is to regard the privacy protection system as a communication model. The methods of privacy information measurement, privacy disclosure measurement, privacy protection intensity quantification and attack capability quantification in different occasions are given preliminarily by defining the concepts of source, destination and channel/introducing information entropy, average mutual information quantity, conditional entropy and conditional mutual information. Although this work only gives a relatively basic information entropy model, it solves the privacy protection system to quantify the problem to establish a viable system foundation. I believe the related research can continue to deepen study. At the same time, as the privacy information has attributes of space-time, subjectivity, fuzziness. The next step I will consider the use of generalized information theory, fuzzy information theory, such as the study of privacy information entropy model.

Acknowledgments

This work is partially supported by Scientific Study Project for Institutes of Higher Learning, Ministry of

Education, Liaoning Province (LQN201720), and Natural Science Foundation of LaioNing Province, China (20170540819). The authors also gratefully acknowledge the helpful comments and suggestions of the reviewers, which have improved the presentation.

References

[1] D. Aritra, C. Min, K. Robert, "Measuring privacy and utility in privacy-preserving visualization," *Computer Graphics Forum*, vol. 32, no. 8, pp. 35–47, 2013.

[2] Y. Chen, *Information Theroy and Coding (in Chinese; 2nd ed.)*, Beijing: Publishing House of Electronics Industry, 2012.

[3] X. Chen, J. Pang, "Measuring query privacy in location-based services," in *Proceedings of the 2nd ACM Conference on Data and Application Security and Privacy*, pp. 49–60, 2012.

[4] A. Deutsch, R. Hull, A. Vyas, *et al.*, "Policy-aware sender anonymity in location based services," in *IEEE International Conference on Data Engineering*, pp. 133–144, 2010.

[5] B. Hoh, M. Gruteser, R. Herring, J. Ban, D. Work, J. C. Herrera, A. Bayen, M. Annavaram, Q. Jacobson, "Virtual trip lines for distributed privacy-preserving traffic monitoring," in *Proceedings of the 6th International Conference on Mobile Systems, Applications, and Services*, pp. 15–28, 2008.

[6] B. Hoh, M. Gruteser, H. Xiong, A. Alrabady, "Preserving privacy in GPS traces via uncertainty-aware path cloaking," in *Proceedings of the 14th ACM Conference on Computer and Communications Security*, pp. 161–171, 2007.

[7] Q. U. Juan, Y. M. Feng, Y. P. Li, L. Li, "An anonymous and provably remote user authentication protocol using extended chaotic maps for multi-server system," *Journal of Shandong University (Natural Science)*, vol. 54, no. 5, pp. 44–51, 2019.

[8] X. Lin, S. P. Li, C. H. Yang, "Attacking algorithms against continuous queries in LBS and anonymity measurement," *Journal of Software*, vol. 20, no. 4, pp. 1058–1068, 2009.

[9] X. Liu, K. Liu, L. Guo, *et al.*, "A game-theoretic approach for achieving k-anonymity in Location Based Services," in *Proceedings IEEE (INFOCOM'13)*, pp. 2985–2993, 2013.

[10] H. Ma, T. Peng, Z. 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.

[11] N. B. M. Mohan, A. S. N. Chakravarthy, C. Ravindranath, "Cryptanalysis of design and analysis of a provably secure multi-server authentication scheme," *International Journal of Network Security*, vol. 20, no. 2, pp. 217–224, 2018.

[12] C. E. Shannon, "A mathematical theory of communication," *Bell System Technical Journal*, vol. 27, no. 3, pp. 379–423, 1948.

[13] A. Ukil, S. Bandyopadhyay, A. Pal, *Privacy Measurement and Quantification*, 2017. (`https://data.epo.org/gpi/EP2919148A1-PRIVACY-MEASUREMENT-AND-QUANTIFICATION`)

[14] C. M. Wang, Y. J. Guo, Y. H. Guo, "Privacy metric for user's trajectory in location-based services (in Chinese)," *Journal of Software*, vol. 23, no. 2, pp. 352–360, 2012.

[15] W. Zheng, Z. Wang, T. Lv, *et al.*, "K-anonymity algorithm based on improved clustering," in *International Conference on Algorithms and Architectures for Parallel Processing (ICA3PP'18)*, pp. 462–476, 2018.

Biography

Hui Xia received his B. S. and M. S. degree from Xidian University, China, in 2003, 2006, respectively. He is currently a associate professor with Shenyang Normal University. His research interests include cloud computing, cryptography and information security.

WeiJi Yang received his B. S. degree from Zhejiang Chinese Medical University, China, in 2005, M. S. degrees from Beijing University of Posts and Telecommunications, China, in 2009. He is currently a Research Associate with ZheJiang Chinese Medical University. His research interests include Information Security and Traditional Chinese Medicine Informatics.

A ZigBee Software Defined Network Security

Alireza Ebrahimi Basabi, Jingsha He, Seyed Mahmood Hashemi, Xinggang Xuan, Muhammad Salman Pathan, and Zulfiqar Ali Zardari

(Corresponding author: Alireza Ebrahimi Basabi)

Faculty of Information Technology[1]
Beijing University of Technology, Beijing 100124, China
Email: Tyler1677@yahoo.com

(Received Nov. 26, 2020; Revised and Accepted Apr. 24, 2021; First Online Nov. 9, 2021)

Abstract

ZigBee is one of the prevalent WSNs and IoT networks. ZigBee devices (ZB-DV), being a part of the wireless network, is susceptible to attacks DDoS that floods the incoming traffic in the net on the devices, this flooding of data traffic originates from several tools or sources that block the entry of new data into the system causing a situation where data cannot be received. The evolving software-defined anything model ensures secure control of IoT devices. In this research work, First, we introduce a general infrastructure for a software-defined ZigBee network (ZBSDN) based on the SDNA and consists of ZBCLs, Switch (ZBSW), Gateway, and End-user ZigBee (ZB). Then we propose an algorithm that can isolate malicious nodes to recognize and decrease DDoS attacks(DDOSATKs) that apply Enhanced sqrt-cosine(ISC) that can isolate malicious nodes.

Keywords: DDoS; Improved Sqrt Cosine Similarity Keywords; The Last Keyword; ZBSDN; ZigBee; ZigBee Software Defined-Network

1 Introduction

ZigBee is a wireless-personal network sensor (WPAN) based on 802.15.4 IEEE standards and a high level networking protocol. ZigBee is a fast, secure, and cost effective system compared to other Wireless Personal Area Networks (WPANs). It was the best choice for a variety of environments, such as smart home automation, healthcare, smart technology, *etc.* with a range of beneficial features of the ZigBee protocol, including low power and reliable networking [17]. There are several software-defined anything in various technical areas, including SDN-defined security, software-defined storage, software defined networks, and software defined data centers [17]. SDN Networking is a modern networking architecture that significantly improves network functionality. SDN distinguishes the control plane and the data plane from decisions taken by a theoretically centralized assistance sharing agent [13]. This decoupling provides a network-wide view of the ZBCL, allowing them to make better-informed decisions [4]. As a result of centralized control, the network becomes more robust, and the network resources managed more efficiently and cost-effectively. After the introduction of this modern networking paradigm, Open-Flow(OpenF) led a regular SDN protocol.

It is a programmable network protocol that governs traffic flows between switches (SW) and routers (RT) irrespective of their unit. The control plane makes choices as to where the packets are to be transmitted, and the data plane makes those arrangements and the following packages. Efficient network transactions can be accomplished by taking advantage of the centralized arrangement-planning of the overall network conditions from a clear and consistent point of view. While SDN offers flexibility and reliable, low-cost control, new weaknesses have been introduced. OpenF is a standard SDN communication protocol for SWs and central controllers. OpenF handles the incoming data packet according to the respective flow information in the SW flow map. If the table is missing from the flow list, the package may be forwarded to the controller for further processing. This allows the attacker to plan a DDOSATK. A large amount of missing packets will be generated by transferring the attacker to the OpenF central controller in the flow table, which will eventually exhaust the central controller and not run the network [20].

We implement this proposed method using the NS2 simulation tool, and the experimental results show that the Improved Sqrt Cosine Similarity method uses the DDOSATK manuals. We've been generating traffic and attack manuals. In our planned ZBSDN system, we evaluate the identification and mitigation of DDOSATKs. The main findings of this article are:

- The extensive configuration for the ZBSDN is defined, and the suggested structure contains of the CL equipment, the SDN-SWs combined with the ZBSDN gateway and the ZigBee equipment. The system equipment originates from the longitudinal CL, which includes the central CL, the underlying CL layer, and the supervision of the supervisor.

- We suggested an ISC algorithm to recognize and decrease the rate of PK-In DDOSATKs. Our simulation conclusion shows that ICS can find nodes that send DDOS packets very quickly and reduce the DDOSATK in ZBSDN.

The remaining part of the paper is separated as follows: Section 2 describes the research study; Section 3discusses ZigBee-Software Defined Network ZBSDN and topologies; Section 4 discusses experimental evaluation and results; Section 5 discusses and concludes future work.

2 Literature Review

Dayal, Neelam *et al.* [17] concentrated on TCAM, logical port, *etc.* These types of attacks could desolate results both through the attack on the controller. Duy, Phan, *et al.* [3] they explored the time factor in the collection of data activities when detecting DDOSATKs in the SDN setting by using entropy metric to presume under-attack status, taking into account variations in the host feature profile.Khan,Samia, *et al.* [19] [9]they prevented black hole and DDOSATKs in MANETs, a lightweight and feasible solution was implemented. The solution proposed will protect the defined path and authenticate the network participating nodes. Wang, Yang, *et al.* [22] they proposed secure the CL aircraft from SDN DDOSATKs and introduced a Safe-Guard Scheme (SGS). Efficient signature modules have been developed: deviation flow identification and dynamic security system. Wang *et al.* [17] Developed an effective HMM-R algorithm matching signature to detect L-DDOSATKs in the data center network and compared to other algorithms such as KNN, SVM, SOM, BP. Myint Oo, *et al.* [20] Introduced identification of DDOSATKs relating SDN-based approach to decrease the authorized user operation and introduced an advanced vector support machine (ASVM) to improve the current backup vector algorithm (SVM) to recognize DDOSATKs. Zhu, *et al.* [18].

Presented SDN-based cross-domain weakness intuition system examined privacy and prevention of DDOSATKs. Baojun Zhou, *et al.* [20] Proposed an online machine-learning DDOSATK intuition applying Spark Streaming to test software set outcomes by machine-learning performance. Wu, Di, *et al.* [14] The proposed algorithm developed an efficient signature scheme PCA to determine the traffic packet data network state. Kothamasu, *et al.* [12] proposed CS DDoS program provides a solution for protecting storage records by classifying incoming packets and making a classification based-decision. The CS DDOS detects and decides if a packet is natural or originating from an intruder during the intuition process. Kokila *et al.* [1] Developed an effective SVM algorithm to identify DDOSATKs. We're named it the NND algorithm for simplicity. The NND algorithm is easy to implement but is unable to manage significant traffic demands. Most specifically, the NND algorithm breaks down DDoS packets based on IP locations that have been compromised by the fake IP locations of the source. It was necessary to implement a backtracking algorithm in order to improve the reply time to find a specific DDoS source. Hameed *et al.* [5] Suggested an integrative mitigation Scheme for DDoS attacks within SDN. Reliable CL to CL protocol has been constructed.

When a CL identifies the severity of a DDoS attack over its system, it stops That Traffic and also informs its neighbors controllers present in different autonomous systems. This aims to mitigate the attack from continuing to spread certain networks and to minimize the attack near its source. The writers were using 3three distinct approaches, Namely optimized, central and mesh, their test bed. The findings demonstrate a quicker DDoS Mitigation plan that occurs in practice time.Deng *et al.* [2]Developed an SDN DosDefender capability using a floodlight controller to detect a DoS attack. Jaafar *et al.* [6] presented a comprehensive report on the recognition of DDoS attacks that used the modern methods used to detect HTTP. The DDoS attack was being presented and a vital research was conducted.S Singh, SKV Jayakumar, *et al.* [15]. proposed a "twin security model" machine learning-based that integrates two knowledge to achieve security quality with regard to detection.

3 Zigbee Software Defined Network

3.1 ZBSDN Architecture

ZBSDN has been massively scalable, transparent, programmable, robust in compliance with policies, and allows connectivity between Devices using ZigBee protocols.It doesn't have the vicinity disadvantages of a peer to peer (p2p) [16] strategy while maintaining its low latency and low-energy advantages. To allow a network service computerization rule for ZigBee devices, we are introducing a programmable software SW requested ZBSW at the edge, which resides as an intermediary Separate the relevant ZBDE to collect all of their p2p transactions. ZBSDN of such SWs is distributed across a wide range of ZB devices and interconnected via the backhaul IP network to address the limited scope of native ZB protocols and to form a wide variety of ZBDEs.

We used Zigbee protocol to illustrate the validity of such an approach to design.Although this protocol is based on a completely different data link layer, it essentially provides a logical support layer (composed of a few protocol layers, as shown in Figure 1 at the top of the underlying link layer. Different protocols use a variety of fields and semantics to support the service layer [11].

Functionalities provided by the ZBSDN architecture are divided into three different planes:

The data plane contains network tools capable of transmitting data according to ZBCLcommands. These devices may be switches, routers, or any other framework provided they support at least one of the southbound in-

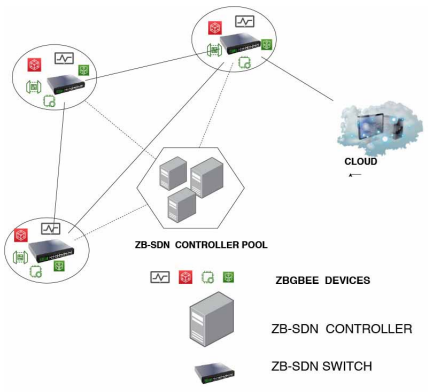


Figure 1: ZBSDN architecture

terfaces and can store the set of rules offered by this interface.Data plane enabling the inspection and retrieval of service layer packets using the match action, counter, rules statistics. ZBCL is represented by the ZBCL(s) that CLs the system in the data plane layer. A management plane that uses on-board devices based on particular link layer protocols.

Two interfaces are used to allow the three layers to communicate:

- Southbound system for aircraft contact with data to be monitored. OpenF is the most prevalent pattern.
- Gateway Northbound?for communication between management and CL systems. It can be implemented using various protocols, including REST (Representative State Transfer), RESTful API, RESTCON, and so on.

3.2 ZBCL Pond Structure

The ZBCL pond incorporated the SDN network into the ZB as a vertical CL structure. It is separated in Copple layers depicted in fig2: the lead CL plate and the central part. The main CL cooperates upwards through the layer application (APP), cooperates downwards by the corresponding CL layer, and connects the infrastructure layer through the central CL layer. Many core CLs, called the central ZBCL, are operated on the primary CL layer by each ZBCL inversely, other key ZBCLs used in the base CL layer as standby CL objects. Core ZBCLs are reliable for managing staff, safeguarding, and preserving the underlying layer of CL and providing access to layer APP networks for northern regions. The Leader decided to use the Paxos algorithm in the main ZBCLs to address the issue of consensus. The Leader will obtain information on the topology of the global network; track the principal ZBCLs, and the primary ZBCL coordinates. Simple ZBCLs are reliable for managing and using resources within the ZB domain. SDN-Devices are managed in the same domain by simple domain CL. The Master ZBCL is connected to the SW and is called the Slave ZBCL. Two ZB systems can communicate across different domains via the same ZBCL.

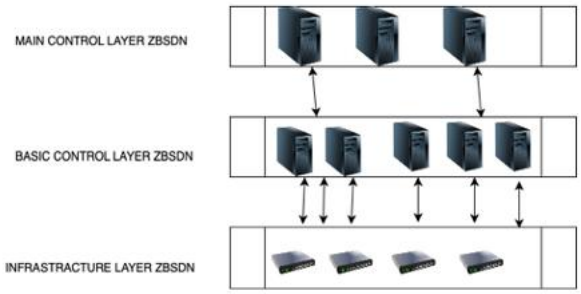


Figure 2: Pond structure

3.3 ZigBee-Devices ZBSDN

Zigbee is the complete IoT networking solution. Provides an layer APP(libraries) for interoperability between different devices and a communication network protocol stack (ZigbeePRO) that uses the IEEE 802.15.4 network layer. The ZigBee protocol splits communication devices into two classes: a Full Function Device (FFD) and a Reduced Function Device (RFD). The FFD has the capability to navigate, while the RFD does not. The coordinator (ZC) and the router (ZR) are part of the FFD within the ZigBee network, while the gateway node (ZED) belongs to the RFD. The FFD can communicate with another FFD and an RFD but the RFD can communicate only with the FFD that set up the network. ZigBee Star Topology is primarily designed for simple communication between a single node and a amount of nodes. The network node is running a hierarchical / node routing system. And the network uses the AODV and Hierarchical / Tree routing hybrid mixed routing approach [7]. The AODV used in ZigBee, however, differs slightly from the AODV used in the ad hoc network. The latter is based on the amount of sequences, which always chooses the newest path. ZigBee AODV(ZBAO) is targeted toward lateral failure [11].ZBDVs, which are part of the wireless network, are weak to DDOSATKs that overwhelm incoming network traffic and this influx of data traffic originates from several tools or sources that block new data entry into the system, leading to a situation where data cannot be received.

3.4 Ddosatks in the Zbsdn Environment

SDN ZBCLs are reliable for unified network intelligence for the operation of the entire network and Zigbee switches (ZB-Sws) are reliable for the transmission of network traffic on the basis of orders from SDN ZBCLs. SDN is currently trying to adapt to changing traffic patterns, high latency and complex APPs, and is an developing technology which could deliver confidence mitigation tools that can quickly detect attacks. Recent studies have shown that SDN is seen as a vital enabler of IoT protection. SDN can simplify predefined IoT rules for devices, involuntarily distinguish and address security weaknesses, configure edge computing and analyze data flow. The DDoS is an obstructed kind of attack in computers on the

network are targeted in a distributed manner from multiple sources, causing in a denial of service to users. With its programmability functionality, SDN offers DDOSATK security solutions that automatically build programs for any complicated scenario. ZBCLs in the ZB processor are reliable for the logical and intelligence-central CL of the entire ZB device in the proposed ZBSDN kernel. Centralized logic CL is informal to handle and customize, And that also triggers security issues. Compared to SDN, ZB-SDN is adept of present proactive systems to identify and prevent DDOSATKs by programming the proposed ZB-SDN kernel. Next section, the DDOSATK protocol for ZBSDN will be checked. The attacker sends a new packet from ZBSDN to a particular SW where the script generates the attack packet. If the flow-table elements of the ZBSW do not fit, the ZBSW encapsulates the packet header in the message(PIM) and passes it to one ZB Controller(ZBCL). While a DDOS produces a huge amount of packets exclusive of similar the ZBCL, the ZBCL obtains a huge amount of packets in messages that not only consumes network resources between the ZBCL and the ZBSW but also includes memory and other ZBCL resources in the CPU, causing in improved latency and even interruption. Also, the ZBSWs are constantly connected to new flow-table items, and the ZBSWs cannot stay to achieve new Obtained packets if the amount of flow-table elements in the ZBSWs reaches the cap. Therefore, the ZBSWs cannot function properly.

3.5 Intuition and Reduction Algorithm of DDOSATK

Due to the wireless transmission medium, WSNs are prone to attacks in the network, which can cause intruders to hack or attack the data being transferred between nodes. WSN are resource-limited DVs with data rates, power consumption and security constraints. It's an advantage for hackers to flood or drain the system's power to block the DV or extract useful information. Intuition and reduction of such attacks [8] is necessary in order to overcome defects and security problems and make ZigBee-SDN a secure, reliable and robust architectural algorithm.

3.6 Problem Description

The ZBCLserves as an SDN brain in the ZBSDN architecture for high-level routing and decision making. The ZB SW forwards packets according to the predefined flow chart of the ZBCL. If the new packet arrives at the ZBSW does not suit the predefined flow table of the ZBCL, the packet in a message transmitted over the secure channel to the ZBCL. DDOSATKs randomly produce a significant amount of stochastic basis IP addresses and target IP address packets. It cannot suit the SW flow chart, which results in the dispatch of a large amount of packets to ZBCL. This can cause the ZBCL to handle channel congestion or exhaust the resource. Yeah, just like Figure 2. Examining the two obstacles the ZBCL faces. The threat referred to in paragraph 1 shall be called a congestion channel protection ZBCL. While the OpenF protocol specifies that each SW and ZBCL has a secure, independent channel, the same physical link is shared by those logically separate CL channels. Logical link congestion causes all other logical links to fail, which leads to the dumping of valid packets.The threat referred to in paragraph 2 shall be called ZBCL land depletion. The information is stored in the SW register after the packet reaches the processor. Nevertheless, information processing includes using ZBCL resources (CPU, memory, bandwidth, *etc.*). When a large amount of packets enter communications, the ZBCL tool is used to avoid transmission of the requirements. Both of these risks may result in the ZBCL failing to provide the service to regular users. In more dangerous situations, the attacker can impersonate the ZBCL during this time to steal user information or launch a significant attack. So efficient, reliable, and scalable methods should be used in the ZBCL to detect high volume DDoS flood attacks.

3.7 Set of Input Port Rate Vectors A and B

At an interval,Δt, of time, we first obtain the PKIN rate αm $(m = 1, 2, \cdots)$. From the input port γ of the SDN-SW boundary, and then we get the set: A=A1, A2, ..., An = α2, α4, α6, ..., α2k, B = B1, B2, Bn = α1, α3, α5, ..., $2k-1$. Where k is the sets or vectors A and B period length. We'll discuss the set of k values in detail in the chapter below.

3.8 Improved ISC Similarity A, B of the Vectors A and B

Improved *Sqrt* Cosine Similarity is a two vector similitude property. ISC similarity is very useful, particularly in the case of Flexible vectors, since only dimensions which are not zero should be noticed. The similarity of the PKIN port b vectors to the ISC can be obtained through the following Equation (1): The similarity between the two vectors is a function of the similarity between the two. ISC similarity is very useful, especially in the case of different vectors, since only dimensions other than zero should be considered. Using the following Equation (1), we can set ISC similarity for δ a, δ b of the port vector rate packet:

$$\begin{aligned} ISC &= (A, B) \\ &= \frac{\sum_{O=1}^{N} \sqrt{\mathbf{A}_O \mathbf{B}_O}}{\sqrt{(\sum_{O=1}^{N} \mathbf{A}_O)} \sqrt{(\sum_{O=1}^{N} \mathbf{A}\mathbf{B}_O)}} \end{aligned} \quad (1)$$

Where $m = 1, 2, k$. Clearly, each the parts of vectors A, B are larger than or match to 0, and therefore we have $0 \leq \delta a, b \leq 1$. Since ISC comparison α a, b is equal to 1, the nearer the vectors angle, the nearer the vectors A and B are to each other. The similarity between the vectors A and B is higher if the ISC similarity αa, b is equal to 1.

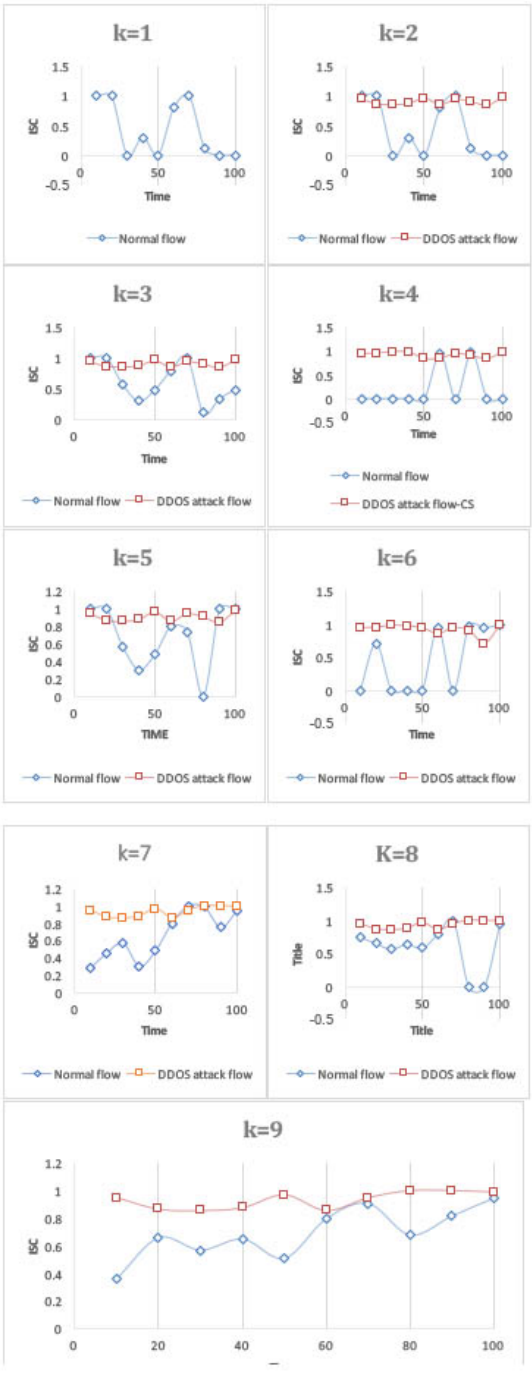


Figure 3: ISC of DDOSATK flow K=1-9

3.9 Deciding If There is a DDOSATK

To evaluate the direction and normal direction of the DDOSATK, the ISC THreshold value (THRVAL) is assumed to be Ω U. If Ω U ≤ δa, b≤1. If (N) the DDOSATK can occur in the SDNSW input port. If 0 ≤ δa, b≤ Ω U, the port data packet may be a regular request. Setting the ISC THRVAL is crucial to the proposed demonstration of the algorithm. Within following section, we address in detail the degree of ISClikeness (D).

3.10 Remove the Packet for the DDOSATK

Multiple DDOSATK flow and normal flow values must be tested to improve the performance protection and accuracy. We get a set of P = δ1, δ3, ..., δl, where l is the amount of Ω U ≤ δ ≤ 1 samples. We use the count to indicate the total value of the amount of DDOSATKer samples and remove the specific kind of data packet in the channel.

3.11 Parameter Settings For Zbsdn

THRVAL ΩU of the ISC and the length k (LK) vectors A and B are two essential elements of the proposed algorithm. When it is too large, several DDOSATK packets are calculated as legitimate demands.

If ΩU is too small, most standard packets considered to be DDOSATKs. The time of intuition is larger if the k value is higher. That could reduce the ZigBee's efficiency and even result in the SDNCLs and SWs being broken down. It can reduce the reliability of the intuition conclusion when the value of k is too low and which conclusively result in a high rate of error. We must conduct a huge amount of experiments and analyses in order to obtain the best possible value of ISC between ΩU and LK.

3.11.1 The THRVAL ΩU Value of the Improved Sqrt Cosine Similarity

interval time Δ0 for transmitting packet attack is unpredictable and The phase scale of 0.005s to 0.05s is 0.005s. The Δ1 period for transmitting daily flow PKIN between 0.05s and 1s and the duration of the Δ1 step is 0.05s. The influence of various k values on ISC can be observed when the value of k decreases from 1 to 9. per k amount, we got 100 ISC costs continuously, and then The ISC mean Figure 4 and standard deviation. Figure 5 can be determined separately.

3.11.2 Length k of Vectors

The mean k 3-5 and standard deviation k 3 to 6 of the ISC among the DDoS flow and normal flow become depicted in Figure 6. If the result of the width k of columns A and

B is changed between 2 to 9, the ISC average is modified. Procedure EQ 3-2 DDOSATK Identify and prevent Procedure for ZB.

Input: The set SW of all border ZBSWs, the amount amount of samples P to fulfill the criterion, the size k of the vectors A and B, the time interim Δ, the amount of PAKINMs,PAKIN and the THRVAL Ω U of the ISC.

Output: The DDoS port that runs and detects a DDOSATK.

3.11.3 Configuring

For both the genetic scheme we initially set to 20s and 300s, respectively, the idletimeout and hardtimeout parameters of usual table flow elements in the ZBSDN ZBCL. Upon recognition of the DDOSATK, idletimeout and hardtimeout become set at 100s and 290s, the same approach for the NNP algorithm, respectively.

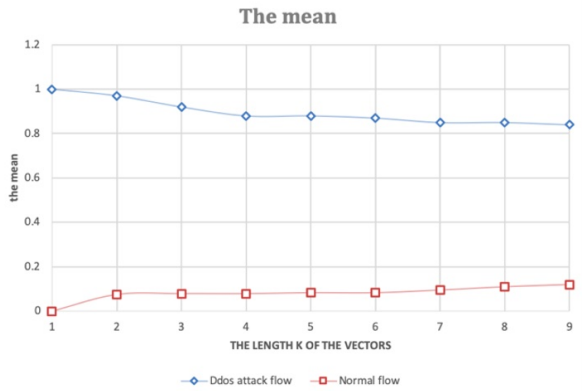


Figure 4: The mean graph for LK vectors

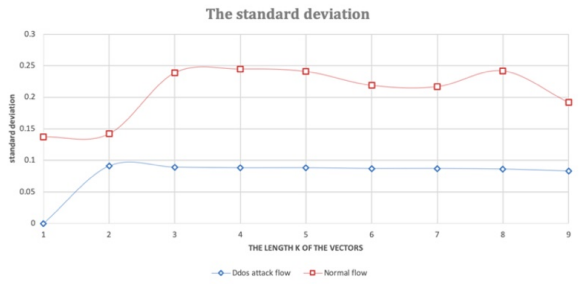


Figure 5: Standard deviation of ISC separately

4 Experimental Evaluation and Results

The algorithm is purposed to identify and decrease DDOSATKs using the ZBSDN kernel where ISC of the PK-IN frequency vectors at the boundary, ZBSDN transfer ports are applied to resolve DDOSATKs that happened in Zigbee DVs. In NS-2 discrete event simulator, this configuration of ZBSDN is simulated. The network is made up of 40 nodes deployed in 200 m. The following experimental criteria checked To test the results purposed algorithm to assess the reliability of the proposed system, ISC of the ZBSDN boundary CL gates PIM frequency vectors used to address whether DDOSATKs occurred on WSN under special ZBSDN conditions. The system is set up using NS2 to build a 40-node mesh topology network that is connected directly to the SW (which might be a terminal or other network).There are 40 nodes and SWs in this network (See Table 1). The NS2 simulator

Table 1: Parametric values for ZBSDN algorithm

Parameter	Value
Simulation time	100 sec
Area	200*200
Amount of nodes	40
Transport layer protocol	TCP
Routing protocol	AODV(ZB-AO)
Antenna type	Omni directional antenna
Network interface	Phy/WirelessPhy
MAC interface	802.15.4
Extended protocol	SDN
CPU Intel-Core?	i5
RAM	16 GB
Storage 320	GB

is based on two programming languages: TCL and C ++ TCL is used to write the simulation scenario and define the DDOSATKer and C ++ to detect and CL the attack.

4.1 Analysis ZBSDN Algorithm

In the following paragraph, we measure the amount of packets throughout messages sent to the zigbee controller(ZBCL) by ZBSDN switches(ZBSWs) and the amount of data packets Obtained by the ZBCLusing both algorithms and raw data.

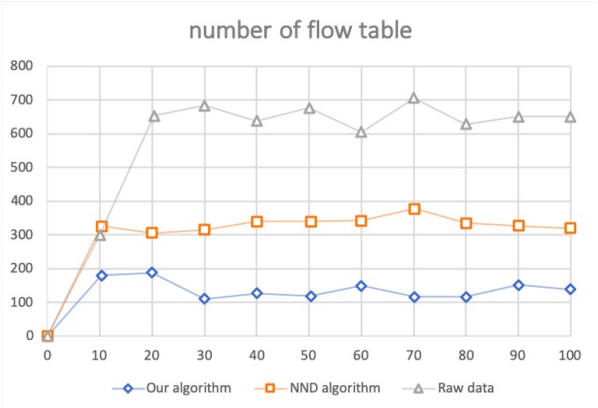


Figure 6: Amount of flow table

Time effect on the amount of packet-in messages Figure 7 (PAKINM) sent to the ZBCL by ZBSW. Over time

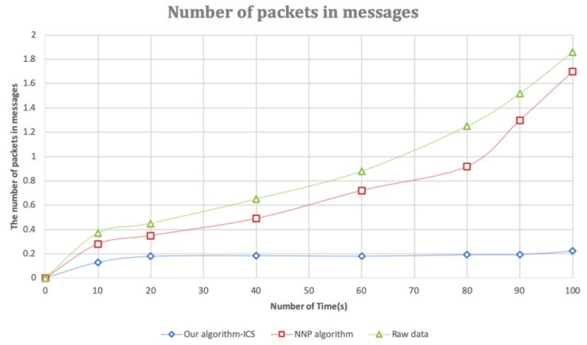


Figure 7: PIMs sent by SDN-SW to the SDN-CL

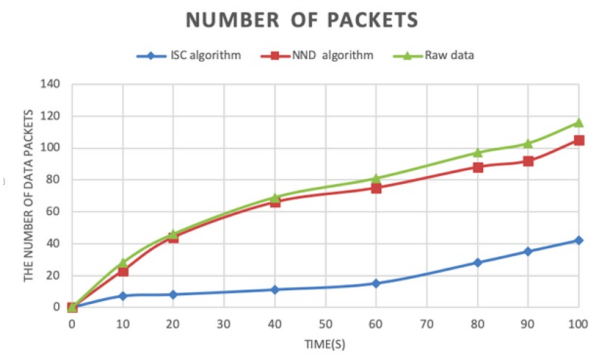


Figure 8: The amount of packets received by the SDN-CL

the amount of PAKINMs sent to the ZBCL by the ZBSW is rising. For our algorithm, the three curves show a slower growth rate than the NND algorithm and the raw data. Our approach applies rules to the stuff in the flow table and removes the packet when a DDOSATK packet is sent to the port using our algorithm. Therefore ZBSW is no longer required to direct a large amount of PAKINMs to the ZBCL. In the case of raw data or the NND algorithm in, the ZBSW quiet desires to drive the ZBCL a large amount of PAKINMs, and therefore the curve slope is very steep.

The effect of time on the bandwidth transition for the ZBCL-channel was shown in Figure 9. When our ZBSDN algorithm detects DDOSATKs, the ZBSW removes DDOSATK packets. Therefore the ZBSDN transfer does not need to give a large amount of PAKINMs Figure 7 to the ZBCL. The value of data packets obtained by the ZBCL gradually decreases to the lowest level until they are stable. The amount of data packets that the ZBCL gets reaches the limit. The raw data Figure 8. Bandwidth transfer to the ZBCL-channel under ZBSDN. Shows steady at 10s, then slowly. The ups and downs that happen emerge because both the DDOSATK flow produces messages for the packet-in and the normal flow. In the case of the NND algorithm [10], the value of data packets obtained after the 20s by the ZBCL is stable and high; this is identical to the raw data after the 20s. Thankfully the algorithm doesn't detect attacks from DDoS. To study, the amount of PAKINMs from the ZBSWs to the ZBCL, and the amount of data packets Obtained by the ZBCL when using our algorithm is lower than the amount of other approaches.

The results show that our algorithm boosts the efficiency of the ZBSDN system under DDOSATKs. ZBSDN routing uses the Ad Hoc, On-Demand Distance Vector (AODV) [21]routing protocol. In the ad hoc network, however, the AODV used in ZBSDN differs slightly from the AODV network. The latter is based on the amount of sequences that always select the newest route. ZBSDN's AODV focuses on route loss. So, for convenience, we are using ZB-AODV to represent ZigBee's AODV.

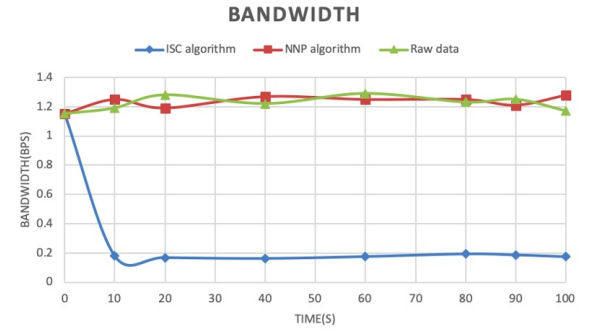


Figure 9: The bandwidth of controller-SW channels in ZBSDN

5 Conclusions

This paper describes the general ZBSDN architecture consisting of the SDN-CL, the SDN-SWs combined with the ZBSDN gateway and the ZigBee system. Instead, with the new ZBSDN architecture, we suggest an appropriate algorithm to intuition and reduction DDOSATKs. At the ZBSDN boundary shift ports suggested algorithm, we reach the THR-VAL of ISC similarity to the packet in frequency vectors; we use the THR-VAL to decide if a DDOSATK happened, to locate the actual DDOSATKer and to block the initial DDOSATK. Finally, the analysis conclusion was that the suggested methodology is capable of finding the ZBSDN network through which a DDOSATK is initiated in a shorter time, managing and reducing the DDOSATK quickly, and currently improving the discovered apparent weaknesses in ZBSDN DVs with computational and memory specifications for ZBSDN DVs. We are contrasting the outcomes of the methods suggested. Future work would suggest and identify such attacks as DDoS based on the Fuzzy Synthetic Assessment model, with other methods of DDoS intuition. Therefore, the ZBCLwill design and implement dynamic routing algorithms and discover more powerful Algorithms based on the ZBSDN method for insight and reduction of DDOSATKs.

References

[1] K. Bhushan and B. B. Gupta, "Detecting DDoS attack using software defined network (SDN) in cloud computing environment," in *The 5th International Conference on Signal Processing and Integrated Networks (SPIN'18)*, pp. 872–877, 2018.

[2] S. Deng, X. Gao, Z. Lu, Z. Li, and X. Gao, "DoS vulnerabilities and mitigation strategies in software-defined networks," *Journal of Network and Computer Applications*, vol. 125, pp. 209–219, 2019.

[3] P. T. Duy, V. H. Pham, *et al.*, "A role-based statistical mechanism for DDoS attack detection in SDN," in *The 5th NAFOSTED Conference on Information and Computer Science (NICS'18)*, pp. 177–182, 2018.

[4] J. Govindasamy and S. Punniakodi, "Energy efficient intrusion detection system for zigbee based wireless sensor networks," *International Journal of Intelligent Systems*, vol. 10, pp. 155–165, 2017.

[5] S. Hameed and H. A. Khan, "Sdn based collaborative scheme for mitigation of ddos attacks," *Future Internet*, vol. 10, no. 3, pp. 23, 2018.

[6] G. A. Jaafar, S. M. Abdullah, and S. Ismail, "Review of recent detection methods for http ddos attack," *Journal of Computer Networks and Communications*, vol. 2019, 2019. (`https://doi.org/10.1155/2019/1283472`)

[7] R. Kallimani and K. Rasane, "Investigation of power consumption in microcontroller based systems," in *Intelligent Communication Technologies and Virtual Mobile Networks*, pp. 404–411, 2019.

[8] S. Khan, F. Hashim, M. F. A. Rasid, and T. Perumal, "Reducing the severity of black hole and ddos attacks in manets by modifying aodv protocol using mac authentication and symmetric encryption," in *The 2nd International Conference on Telematics and Future Generation Networks (TAFGEN'18)*, pp. 109–114, 2018.

[9] M. Latah and L. Toker, "Artificial intelligence enabled software-defined networking: A comprehensive overview," *IET Networks*, vol. 8, no. 2, pp. 79–99, 2018.

[10] Y. Liu, M. Dong, K. Ota, J. Li, and J. Wu, "Deep reinforcement learning based smart mitigation of ddos flooding in software-defined networks," in *IEEE 23rd International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD'18)*, pp. 1–6, 2018.

[11] D. Loghin, S. Cai, G. Chen, T. T. A. Dinh, F. Fan, Q. Lin, J. Ng, B. C. Ooi, X. Sun, Q. T. Ta, *et al.*, "The disruptions of 5g on data-driven technologies and applications," *IEEE Transactions on Knowledge and Data Engineering*, vol. 32, no. 6, pp. 1179–1198, 2020.

[12] C. E. Mihanjo, "Detection of DDoS attacks and flash events occuring simultaneously in network traffic using deep learning techniques," *Information and Communication Technology*, 2020. (`http://hdl.handle.net/20.500.12661/2647`)

[13] M. M. Oo, S. Kamolphiwong, T. Kamolphiwong, and S. Vasupongayya, "Advanced support vector machine-(ASVM-) based detection for distributed denial of service (DDoS) attack on software defined networking (SDN)," *Journal of Computer Networks and Communications*, vol. 2019, 2019.

[14] A. Sahi, D. Lai, Y. Li, and M. Diykh, "An efficient DDoS TCP flood attack detection and prevention system in a cloud environment," *IEEE Access*, vol. 5, pp. 6036–6048, 2017.

[15] S. Singh and S. K. V. Jayakumar, "Twin security model—a machine learning-based approach for DDoS attack detection in SDN," in *International Conference on Soft Computing and Signal Processing*, pp. 53–62, 2019.

[16] M. Uddin, S. Mukherjee, H. Chang, and T. V. Lakshman, "SDN-based multi-protocol edge switching for iot service automation," *IEEE Journal on Selected Areas in Communications*, vol. 36, no. 12, pp. 2775–2786, 2018.

[17] W. Wang, X. Ke, and L. Wang, "A HMM-R approach to detect L-DDoS attack adaptively on SDN controller," *Future Internet*, vol. 10, no. 9, pp. 83, 2018.

[18] D. Wu, J. Li, S. K. Das, J. Wu, Y. Ji, and Z. Li, "A novel distributed denial-of-service attack detection scheme for software defined networking environments," in *IEEE International Conference on Communications (ICC'18)*, pp. 1–6, 2018.

[19] J. Xie, F. R. Yu, T. Huang, R. Xie, J. Liu, C. Wang, and Y. Liu, "A survey of machine learning techniques applied to software defined networking (SDN): Research issues and challenges," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 1, pp. 393–430, 2018.

[20] B. Zhou, J. Li, J. Wu, S. Guo, Y. Gu, and Z. Li, "Machine-learning-based online distributed denial-of-service attack detection using spark streaming," in *IEEE International Conference on Communications (ICC'18)*, pp. 1–6, 2018.

[21] R. Zhou, J. Lin, L. Liu, M. Ye, and S. Wei, "Analysis of SDN attack and defense strategy based on zero-sum game," in *International Conference on Brain Inspired Cognitive Systems*, pp. 479–485, 2019.

[22] L. Zhu, X. Tang, M. Shen, X. Du, and M. Guizani, "Privacy-preserving ddos attack detection using cross-domain traffic in software defined networks," *IEEE Journal on Selected Areas in Communications*, vol. 36, no. 3, pp. 628–643, 2018.

Biography

Alireza Ebrahimi Basabi is a PhD student at Beijing University of Technology under the supervision of Professor JingShaHe at Beijing University of Technology (BJUT). He has a background in system

administration and software development. His research interests include social media, security routing, IOT (Internet of Things), Ai (Artificial Intelligence), cloud computing, information security and the network.

Jingsha He received his Master's and doctoral degrees in computer engineering from the University of Maryland at College Park in the US. He is a professor in the School of Software Engineering at Beijing University of Technology (BJUT) in Beijing, China. Prior to joining BJUT in 2003, Prof. He worked for several multi-national companies such as IBM Corp., MCI Communications Corp. and Fujitsu Labs in the US. where he published more than 10 papers and received 12 U.S. patents. Since joining BJUT in 2003, Prof. He has published nearly 240 papers in journals and international conferences, received nearly 40 patents and 30 software copyrights in China and co-authored 7 books. He has been the principal investigators of more than 20 research projects. Prof. He's research interests include information security, wireless networks and digital forensics.

Seyed Mahmood Hashemi get his BSc and MSc degrees from Islamic Azad University. Now, He is Recent PhD graduate worked under the supervision of Professor JingShaHe in the Beijing University of Technology (BJUT). His research interests include social media, security Routing, IOT (Internet of Things), Ai (Artificial Intelligence), cloud computing, information assurance and Network.

Xinggang Xuan is a PhD student of Beijing University of Technology and the main research direction is for information security, wireless sensor networks.

Muhammad Salman Pathan received his PhD the Faculty of Information Technology, Beijing University of Technology, China. He received his B.E. and M.E degree from Mehran University of Engineering and Technology, Pakistan in 2011 and 2014 respectively. Currently, he is doing Post Doc at the Faculty of Information Technology, Beijing University of Technology, China. His research interest is Wireless Communications, Information Security, sensor, network security.

Zulfiqar Ali Zardari received his B.E. and M.E degree from Mehran University of Engineering and Technology Jamshoro in Sindh, Pakistan 2011 and 2015 respectively. Currently, he received his PhD the Faculty of Information Technology Beijing University of Technology, China. He has published more than 10 research papers as a first and co-author in international journals. His research interest area is Mobile ad hoc networks, Wireless Communications, Information Security, sensor network security, Computer Networks and Network Security. .

Detect Cross-Site Scripting Attacks Using Average Word Embedding and Support Vector Machine

Fawaz Mahiuob Mohammed Mokbal[1,2], Dan Wang[1], and Xiaoxi Wang[3]
(Corresponding author: Dan Wang)
College of Computer Science, Faculty of Information Technology, Beijing University of Technology[1]
Beijing, Beijing 100124, China
Email: wangdan@bjut.edu.cn
Faculty of Computer Science, ILMA University, Karachi 75190, Pakistan[2]
State Grid Management College, State Grid Management College, Beijing, China[3]
(Received Nov. 26, 2020; Revised and Accepted May 6, 2021; First Online Nov. 5, 2021)

Abstract

Web applications are still the preferred target for cybercriminals, as these applications have achieved widespread popularity among individuals and companies. The cross-site scripting attack (XSS) is one of the most severe concerns highlighted at the forefront of information security experts' reports. In this study, we proposed the NLP-SVM method to detect web-based XSS attacks. The method used Natural Language Processing (NLP) for processing text payload attacks and the SVM model for the detection task. The XSS attack payloads were converted into vectors at payload-level instead of word embedding-level. Subsequently, the generated vectors were used for modeling using a support vector machine (SVM) algorithm. The proposed method has successfully surpassed double-check tests include the 10-fold cross-validation approach and the hold-out dataset approach. Extensive analyses of the results determine that the proposed method can efficiently and effectively detect the XSS-based attacks with minimum FN and FP rates. Furthermore, the proposed method had many worthy advantages over the well-known eight algorithms that used the same data. It achieved promising and state-of-the-art results with accuracy, precision, detection probabilities, F-Score, FN rate, FP rate, Misclassification, and AUC-ROC scores of 99.44%, 99.54%, 99.64%, 99.59%, 0.4%, 1.0%, 0.56%, and 99.33%, respectively.

Keywords: Attack Detection; Cross-Site Scripting attack (XSS); Embedding Vectors; Natural Language Processing (NLP); Support Vector Machine (SVM); Web Application Security

1 Introduction

With the development of technology and the Internet's availability everywhere, web applications have occupied a prominent place among the people, and a growing interest by profit and non-profit organizations. Simultaneously, cybercriminals' desire has increased to target these applications to obtain informational or financial gains. According to the Common Vulnerability Scoring System (CVSS), the overall number of new vulnerabilities increased by 17.6% (from 17,308 in 2018 to 20,362 in 2019) [16]. Consequently, this increase represents the keen security concern for various applications, Notably, those carried out in high priority facilities or high-availability operations, such as medical care, banking, e-commerce, *etc.* [13].

One of the most concerned and high-risk cyber-attacks on web applications is Cross-Site Scripting (XSS). The cybercriminals exploit vulnerabilities in the web application to inject their malicious code into the HTML pages, typically in script form executed on users browsers. The cyber-criminals can then extract personal information or steal user cookies to hijack an identity in a fraud session. Consequently, They can steal confidential data or even take hold of those devices [13, 15, 22].

Recently, XSS vulnerabilities continued to grow and ranked as the second most common vulnerability at 14% among 2018 web application vulnerabilities, and they still the second most common vulnerability in 2019. Furthermore, the XSS vulnerabilities were the most common by 44.6% (703) in 2019 of all the 1530 WordPress vulnerabilities [17]. Additionally, XSS enlisted as the most attack vectors, nearly 40% of all attacks used against web applications in 2019, as per PreciseSecurity research [18].

Despite many tools existing to check malicious code existence and the awareness of these vulnerabilities, their

number will not decrease in the future. The immediate impact of the exploit of these vulnerabilities, as well as the lack of preconditions required to exploit them in most cases, could be the main reason for this.

Many researchers in the web security domain have used different techniques such as traditional-based and machine learning-based to mitigate and detect this type of cyber-attack, operating them on the server-side, client-side, or both [27]. However, they all still face some limitations present to

1) Have remarkable missing cases, that is, false negative (FN) rate;

2) Produce an unignorable issue of false positive (FP rate).

The FN issue is crucial since the real threat will pass the system undetected, leading to a complete penetration of the system, losing a lot, such as reputation or financial point of view. In contrast, the FP problematic will cause the legitimate user to be deprived of the service. Further, a heavy burden will add to the security system specialists that will investigate, waste much time, and lose user's confidence and, therefore, companies losing money.

This study proposed the NLP-SVM method to detect web-based XSS attacks to overcome the FN and FP issues, which is highly efficient and has the tendency to defeat such cyber-attacks. The proposed method processes the attacks text payload using Natural Language Processing (NLP). It provides the attack vectors at payload-level instead of word embedding-level to fit the support vector machine algorithm for modelling. More than 20200 samples of XSS text payloads were used, and the method was tested using double-check. that is, our method was tested using 10-fold cross-validation and hold-out testing dataset. Extensive analyses of the results determine that the proposed method able to detect the XSS-based attacks efficiently and effectively with minimum FN and FP. Furthermore, the proposed method had many worthy advantages compared to eight algorithms that used the same data. It achieved promising and state of the art results with accuracy, precision, detection probabilities, F-Score, FN rate, FP rate, Misclassification, and AUC-ROC scores of 99.44%, 99.54%, 99.64%, 99.59%, 0.4%, 1.0%, 0.56%, and 99.33%, respectively.

The rest of this study is organized as follows: Section 2 discusses the most recent related work. Section 3 presents the essential details about this study's methodology and techniques. While Section 4 offers the experimental design and evaluation mechanism strategy, including in-deep details about results, discussion, and comparing the proposed with different eight algorithms. Finally, section 5 concludes this study with its significance and highlights future research directions.

2 Related Work

Many researchers using traditional-based XSS attack mitigation including pattern filtering [28] sanitization-based [2], browser extension [11, 23, 26], proxy-based [8] signatures of script code [21], and Content security policy (CSP) [6]; More details can be find in [20]. These traditional methods are still facing some limitations, especially in the FP rate. The researchers have been moved to a machine learning technique to improve such an attack's detection rate.

In the study presented in [1] the authors used the word2vec tool to determine the occurrence frequency and coincidence of XSS scripts payloads. The occurrence frequency vectors are used for modelling using different machine learning models. However, using the frequency of occurrence and coincidence resulting in large and sparse vectors (mostly 0 values) describes script but not the meaning of the words.

The dynamic feature extraction from content and integrates it with Artificial neural networks (MLP) for the detection task has been proposed in [13]. However, the detection rate of 98.35% needs to improve.

A study proposed by the authors in [4] presents the Browser-based method as defenses against XSS attacks using tokens authentication. Their method hypothesis was based on 2464 cookies gathered from 215 ranked websites; Then, the problem was formulated as a binary classification, and different ML algorithms were used for classification. However, using only tokens authentication is not enough to defeat such attacks. Further, the best detection rate obtained with a random forest model was 83%. In [25], the authors proposed a model for malicious code detection using the stacked denoising autoencoder technique, which resulted in big dimensions, forcing the authors to use sparse random projections for dimensions reduction. However, the FP and the detection rate were 4.2% and 93%, respectively, which is inadequate for detecting malicious attacks.

In the study [19], the authors proposed the XSS attacks classification model for social sites. They applied various algorithms that trained on 100 samples only. However, the dataset is too small and a detection rate score of 0.949.

Wang *et al.* [24] introduced a hybrid analysis method to detect malicious web pages. They used three gropes of features includes URL, HTML, and JavaScript, to classify malicious pages. The result of the detection rate was 88.20%, which is inadequate for detecting malicious attacks.

3 Detection Methodology

3.1 XSS Payload-Level Vector

Since the XSS payloads data are usually in script form, machine learning requires that we first represent the text numerically. The straightforward bag-words model encoding schemes such as frequencies and word counts gen-

Table 1: Examples of payload attacks in the dataset

No	Payload	Class
1	<link rel=import href=data:text/html,<script>alert(1)</script>	XSS
2	&";!–>XSS <=&{()}	XSS
3	<SCRIPT>alert('XSS');</SCRIPT>	XSS
4	<svg xmlns=http://www.w3.org/2000/svg" onload="alert(document.domain)"/>	XSS
5	<script/src=data:,eval(atob(location.hash.slice(1)))//#alert(1)	XSS
6	<a href="/wiki/Social_software" title="Social software"¿Social information systems 	XSS
7	Dreyfus' critique of artificial intelligence :	XSS

erate huge and sparse vectors. Therefore, the word vectors also called the word embeddings technique is introduced to represent each word numerically so that the vector corresponds to how that word is used or what it means. Word embeddings are learned by considering the context. That is, getting the meaning of words from their appearance in context. The word vector approach further improves straightforward bag-words model encoding schemes. The words that appear in similar contexts will have similar vectors. However, using this method resulting in 94-dimensional for each word. Since we do not have word-level labels and only have XSS attack payload-level labels, ML models won't be able to use the word-level embedding. Therefore, the vector representation at payload-level for each attack sample is needed. To deal with this issue, we averaged each word's vectors (word-level) for each attack payload and obtained a single vector at payload-level for each sample. Then, these vectors are used as input for modelling. The NLP is used to pre-processing the XSS attack payload. In particular, we used the current and efficient framework in NLP called spaCy [9]. It is a free and open-source library for advanced (NLP) in Python specifically built for production use and helps create applications that comprehend and process large volumes of text. It can create knowledge extraction, natural language comprehension systems, and pre-process text for ML or Deep learning.

spaCy provides embedding learned from a model called Word2Vec [12] and can calculate the average XSS attack vectors obtaining by doc.vector class. These attack vectors are passed to scikit-learn models. The steps of our method are presented with (Algorithm 1).

Algorithm 1 NLP-SVM steps

1: Begin
2: Initial dataset upload:
3: Initialize the *observations storage.*
4: **for** each XSS text payload in the dataset **do**
5: Tokenized payload into the sequence of token
6: **for** each token **do**
7: Get the vector representation
8: **end for**
9: Calculate the average vectors for each token in the payload
10: Combine all the tokens vectors into a single payload vector
11: Append payload vector with its label
12: **end for**
13: Use payload vectors as input to training ML models
14: Validation ML using the 10-fold cross-validation method
15: Test the fully training model on a hold-out data set
16: End

3.2 Collection XSS Dataset

The XSS payload attacks consist of 3944 samples were collected from PortSwigger Research [17] and Github repository [7]. The benign payloads consist of 6313 samples were gathered from [10]. Table 1 present a few payload samples.

Looking at sample No.1 in Table 1 as an example, when the XSS payload is converted into a word vector, we got a vector for each word, each vector with 96-dominations. That is 27 vectors, each with 96-dimensions, making machine learning model inability to learn. Furthermore, the payloads within the dataset do not have word-level labels and only have the payloads-level labels. Therefore, each payload's vector representation is obtained using word's vectors' average and used the labels at payload-level. Later, we obtained 20257 vectors, each with 96-dimensions for the entire dataset. Table 2 shows the word level vectors and the payload-level vector dimensions.

3.3 Machine Learning Model

Although machine learning has gained a prominent place in cybersecurity, most XSS-based attack detection models still suffer some limitations. Precisely, there are limitations in low detection rate, high false-positive alerts, or high false-negative alerts. Therefore, machine learning models need clean and accurate data to be able to detect attacks efficiently. One of the most popular algorithms is the support vector machine (SVM). SVM is a robust supervised learning algorithm used for classification, re-

Table 2: Payload attack tokens with word-level and payload level vector-dimensions

Payload	**Payload tokens**	**World victors-dimensions**	**Payload vector-dimensions**	**Entire dataset dimensions**
<link rel= import href=data:text/html,<script> alert(1)</script>	<, link, rel= import, href, = data, :, text, /, html, &comma,;, &lt,;, script, &gt,; alert(1),&lt,; &sol,; script, &gt,;	(27, 96)	(1, 96)	(20257, 96)

gression, and outlier detection [22].

Given training vectors $x_i \in \Re^n, i = 1, ..., N$, and their corresponding labels from two classes $y_i \in \{-1, 1\}, i = 1, ..., N$, the classification problem is formulated as Equation (1).

$$y_i(x) \quad = \omega^T \phi(x) + b. \tag{1}$$

Where ϕ is the feature-space transformation function, w is vectors and b is the linear classification bias.

The SVM objective is to find the optimal hyper-plane $w \in R^n$ in N-dimensional space and distinctly classify the data points by maximizing the nearest positive and negative samples' margin. This procedure is expressed in Equation (2):

$$\min_{wb} \max \quad = \frac{1}{2} \; subject\; to : y_i(x) \quad = \omega^T \phi(x) + b. \tag{2}$$

However, calculating $\phi(x)$ is very ineffectual and could be impossible because of the introduction of the Lagrange multipliers $\alpha = \{\alpha_i\}, i = 1, ..., N$. Therefore, the former minimization problem is converted into a maximization problem [5]. Further, the feature space can be high-dimensional and may have infinite dimensions. The kernel function is introduced to implicitly define the feature space and efficiently compute very high dimensional spaces. Furthermore, the controller parameters, also known as soft-margin, that allow the violation of the margin constraint are introduced to solve the optimization problem. The kernel function with its parameters is shown in Equation (3). In our experiment, we used the Radial basis function (RBF) defined in Equation (4).

$$\begin{gathered} \max_{\alpha} D_\gamma \; (\alpha) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{i=1}^{N} \alpha_i \alpha_j y_i y_j \kappa_\gamma \; (x_i x_j), \\ subject to := \begin{cases} \sum y_i \alpha_i = 0 \forall i \\ 0 \leq \alpha_i \leq C \forall i \end{cases} \end{gathered} \tag{3}$$

where κ_γ refers to the RBF function kernel is defined as in Equation (4), and C refers to a regularization term that controls the allowed misclassification-level for the training samples. The $\max_\alpha D_\gamma \; (\alpha)$ is the quantity of an upper bound on misclassification probability of kernel κ_γ .

$$\kappa_\gamma \; (x, y) \quad = \exp^{-\gamma^{(x-y)^2}} \tag{4}$$

We tuned γ parameter (gamma) to 0.01 and C parameter to 5 based on practical experiments.

3.4 Experimental and Evaluation

3.5 Dataset Subdivisions

The dataset consists of 20,257 text payload samples, in which 6,313 are benign, and 13,944 are malicious. The dataset has been split randomly and separately into two parts with a ratio of 70%: 30%: for training and testing sets, respectively. The details of the complete dataset are shown in Table 3.

Table 3: Dataset subdivisions

Name	**Benign**	**Malicious**	**Total**
Training dataset	4392	9787	14179
Hold-Out dataset	1921	4157	6078
Total dataset	6313	13944	20257

3.6 Performance Evaluation Metrics

In this research, accuracy, precision, detection rate (DR), Error Rate, false positive, false negative, F-score, and ROC-AUC curves are selected to evaluate the performance of the proposed scheme. These measurements are based on confusion matrix [14]. The TN represents whether the normal case is correctly classified as normal or not. FP or type I error means the normal cases that are incorrectly labeled as an XSS attack. FN or type II error represents an XSS payloads attacks that are incorrectly identified as normal. TP means that an attack payload is correctly identified as an attack. The detailed derivation of the selected performance metrics are shown in the following equations:

$$\begin{aligned} Precision \quad &= \left(\frac{TP}{TP + FP} \right) \\ Detection_{Rate} \quad &= \left(\frac{TP}{TP + FN} \right) \\ TP_{Rate} \quad &= \left(\frac{FP}{TN + FP} \right) \\ FN_{Rate} \quad &= \left(\frac{FN}{TP + FN} \right) \\ F - score \quad &= 2 \left(\frac{Recall \times precision}{Recall + precision} \right) \end{aligned}$$

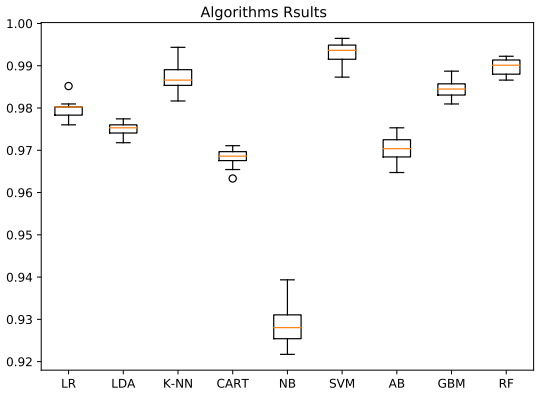


Figure 1: Comparison of algorithms results

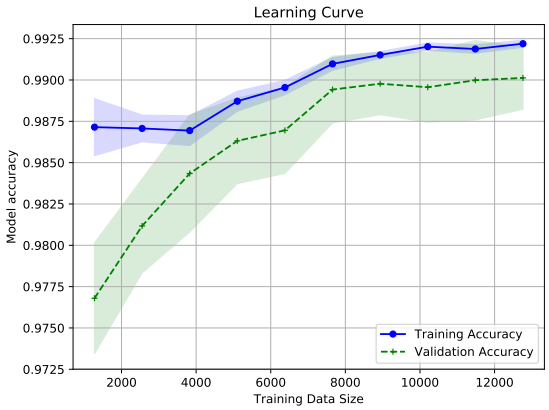


Figure 2: NLP-SVM learning and validation curves with 10-fold

$$Misclassification_{Rate} = \left(\frac{FP+FN}{TP+TN+FP+FN}\right)$$

$$ROC-AUC = \frac{1}{2}\left(\frac{TP}{TP+FN}\right)+\left(\frac{TN}{TN+FP}\right)$$

3.7 Results and Discussion

In this study, SVM alongside eight ML algorithms are implemented, including Logistic Regression (LR), Linear Discriminant Analysis (LDA), K-Neighbors (KNN), Decision Tree (CART), GaussianNB (NB), AdaBoost (AB), Gradient Boosting (GB), and Random Forest (RF) [3]. The entire experiments were done on the operating platform LinuxMint-19-tara, 16 GB RAM, Intel Xeon CPU E-5-2620 v3@ 2.40GHz, GPU NIVIDA (Quadro K220). SpaCy version is V2.0, and the Python version is 3.6.7. For strict validation of our model's proposal, we used double-check. All models are trained and tested using 10-fold cross-validation at the first check. Then the Hold-Out test set is used to assess the performance of the final and fully trained models. Overall, the results achieved by all the algorithms were promising. However, the SVM model achieves superior results in both 10-fold cross-validation and the hold-out dataset test. Figure 1 shows that the SVM performance comes in the first position, within an accuracy of 0.9929 and 0.0028 standard deviation. Followed by RF, KNN, GBM, LR, LDA, CART, and NB score of 0.989632, 0.987023, 0.984484, 0.979759, 0.975034, 0.968193, and 0.929121, respectively.

The NLP-SVM model then was tested under 10-fold cross-validation. The SVM parameters of gamma and C were tuning to 0.01 and 5, respectively. The NLP-SVM testing result demonstrated the ability and effectiveness of the model to detect the XSS payloads. Figure 2 shows the NLP-SVM learning and testing curve. The convergence and smoothness between the learning curve and the verification curve indicate that the model learns very well, especially after 4000 sample size.

To clarify the NLP-SVM discriminative robustness, we evaluated it with the ROC curve (receiver operating characteristic). The ROC curve is a crucial measure for any classification model's performance that visualizes classification efficiency and provides a full sense of its performance. Verification of NLP-SVM was applied at fold-level, and the mean of 10-fold was calculated. The Figure 3(a) and 3(b) shown the ROC curves performance of NLP-SVM. The mean area under the ROC curve was 0.99 with ?00, proving the model's robustness and ability to detect XSS attacks.

Furthermore, The NLP-SVM model was evaluated by a score and scalability functions. Figure 4 shows the model's results evaluation consisting of (A) Score Vs. Training samples, (B) Scalability Vs. Fit-times, and (C) the performance Vs. Training samples. The results clear that the model has a powerful ability to detect XSS-based attacks. All evaluation results are very consistent and harmonic, which is another strong proof of our proposal outstanding performance.

The second evaluation was done by testing all models on the hold-out dataset. This process was performed after all models were training. The confusion matrix of all models is shown in Table 4. We provided in-depth details with extensive evaluation measurements to compare all the models' results in Table 5. Although the performance of all algorithms was efficient, there are some different crucial points. Compared to SVM, we can see that some models are selective behave. They have higher precision than the detection rate, resulting in an increased FN rate, which means that the real threats will pass through the system undetected. The LR, LDA, CART, NB, AB, and GBM are examples of this category. Another observation is that some models have low precision and low detection rate, resulting in increased FP and FN simultaneously. The NB and CART are examples of this category.

Furthermore, some models have difficulty distinguishing between benign and attack samples where the XSS attack class results are very well, but the benign class results are not. Therefore, they classify many benign samples as attacks, leading to an increasing FP rate. The RF, GBM, AB, NB, CART, LDA, and LR are Fall in this category.

On the other hand, the proposed model is an ideal

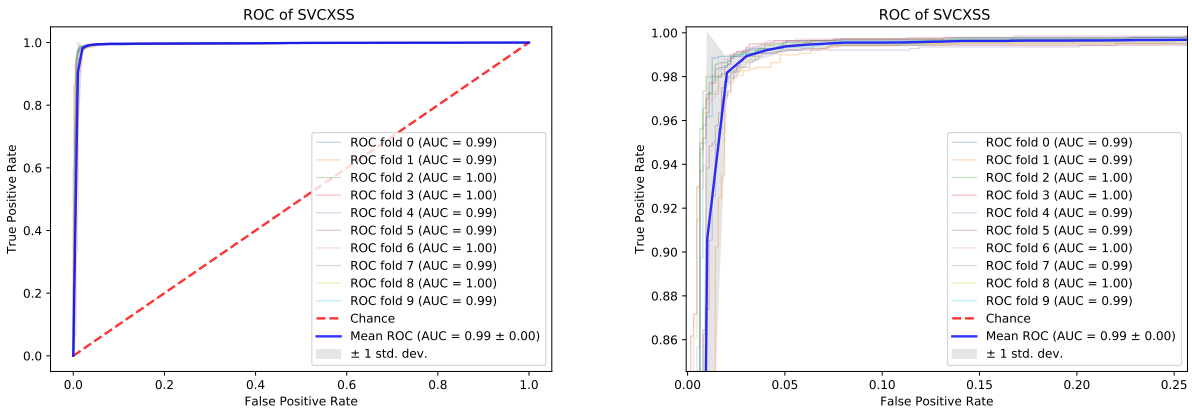


Figure 3: The ROC curves performance of NLP-SVM. (a)SVM model ROC curve for each fold and the mean with a standard deviation, (b) SVM model ROC curve for each fold and the mean with a standard deviation Zoning on the top left.

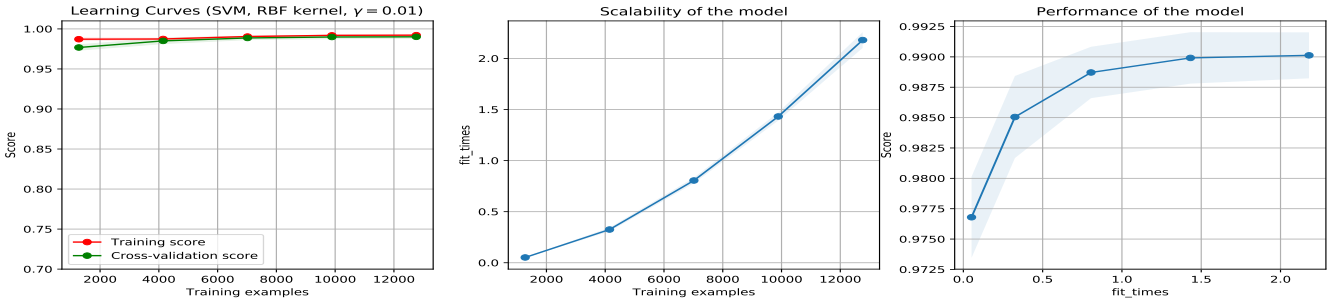


Figure 4: (A) Model learning curve score, (B) show the scalability curve of the model, and (C) show the Model performance curve

model by all measures, which have minimal false negatives and false positives simultaneously. The false negatives score is 0.004 (0.4%), and the false positives score is 0.010 (1.0%), as shown in Figure 5. This ideal result is crucial in such security systems. The FNs are posing real threats to the system. They may lead to a complete penetration of the system, losing a lot, such as reputation or financial point of view. Therefore, it must be taken into consideration very well.

Due to the FP, the legitimate user may deprive of the service. A heavy burden will add to the security system specialists that will investigate, waste much time, and lose users and money's confidence. Therefore, the FP must be reduced to the maximum extent. Other pointers to our proposed model's robustness are the ROC carve and F-score measurements. The area under the ROC curve reaches 99.33%, and the F-score to 99.35%, reflecting the harmony of accuracy and recall and implies the model's power.

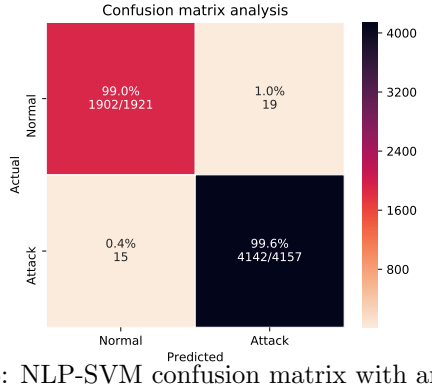


Figure 5: NLP-SVM confusion matrix with analyzing

Table 4: Confusion matrix of all models on hold-out dataset

Model	TP	FN	TN	FP	Total Samples
LR	1841	48	4109	80	6078
LDA	1840	67	4090	81	6078
K-NN	1903	60	4097	18	6078
CART	1816	95	4062	105	6078
NB	1616	136	4021	305	6078
AB	1820	74	4083	101	6078
GBM	1855	37	4120	66	6078
RF	1881	24	4133	40	6078
SVM	1902	15	4142	19	6078

3.8 Comparison with Previous Works

This section compares our work with the reported results of three previous related works in [1, 19, 24], as shown in Table 6. The studies we compare our work with were discussed in more detail in the related work section.

The comparison results demonstrate our proposed work performance is better than others in most measurement. Notably, NLP-SVM's detection rate is the highest, which is critical in such attacks detection systems. Furthermore, we assessed our proposed work on various performance metrics not mentioned in the relevant works.

4 Conclusions

This study presents NLP-SVM model using average word embedding method to detected web-based XSS attacks. Our proposal is used NLP for processing text payloads attacks and the SVM model for the detection task. The detection model has been proved efficient to achieve higher accuracy and a remarkable detection rate with minimal False Positive and Negative rates. The NLP-SVM model adopted a large dataset for training and testing. Numerous analyses have been performed to test the proposed model at various stages. The experimental results confirm the efficiency and idealism of the NLP-SVM method with significant perfection and harmony of performance on multiple measurements of both classes, compared to eight ML algorithms. Moreover, the proposed model outperforms all models in all aspects.

Although our method has proven highly efficient in detecting such attacks, the attack payloads should be extracted from the content to detect the attack automatically. This task will be our future work, as we plan to propose a mechanism for the automatic extraction the attack payload from the content and integrated with NLP-SVM model.

References

[1] S. Akaishi and R. Uda, "Classification of xss attacks by machine learning with frequency of appearance and co-occurrence," in *The 53rd Annual Conference on Information Sciences and Systems (CISS'19)*, pp. 1–6, 2019.

[2] P. Biswajit, G. Tyler, and M. Priyanka, "Handling cross site scripting attacks using cache check to reduce webpage rendering time with elimination of sanitization and filtering in light weight mobile web browser," in *First Conference on Mobile and Secure Services (MOBISECSERV'15)*, pp. 1–7, 2015.

[3] G. Bonaccorso, *Machine Learning Algorithms: Popular Algorithms for Data Science and Machine Learning*, 2018. ISBN: 1789347998.

[4] S. Calzavara, G. Tolomei, A. Casini, M. Bugliesi, and S. Orlando, "A supervised learning approach to protect client authentication on the web," *ACM Transactions on the Web (TWEB'15)*, vol. 9, no. 3, pp. 1–30, 2015.

[5] M. U. Diwekar, "Introduction to applied optimization," *Springer Optimization and Its Applications*, vol. 22, 2020.

[6] I. Dolnák, "Content security policy (CSP) as countermeasure to cross site scripting (XSS) attacks," in *The 15th International Conference on Emerging eLearning Technologies and Applications (ICETA'17)*, pp. 1–4, 2017.

[7] GitHub.com. *Cross Site Scripting (XSS) Vulnerability Payload List*, 2020. (`https://github.com/payloadbox/xss-payload-list`)

[8] S. Goswami, N. Hoque, D. K. Bhattacharyya, and J. Kalita, "An unsupervised method for detection of xss attack," *International Journal of Network Security*, vol. 19, no. 5, pp. 761–775, 2017.

[9] M. Honnibal I. and Montani, "SpaCy 2: Natural language understanding with bloom embeddings, convolutional neural networks and incremental parsing," *To Appear*, vol. 7, no. 1, pp. 411–420, 2017.

[10] Kaggle.com. *Cross Site Scripting XSS Dataset for Deep Learning*, 2020. (`https://www.kaggle.com/syedsaqlainhussain/cross-site-scripting-xss-dataset-for-deep-learning`)

[11] B. Mewara, S. Bairwa, J. Gajrani, and V. Jain, "Enhanced browser defense for reflected cross-site scripting," in *Proceedings of 3rd International Conference on Reliability, Infocom Technologies and Optimization*, pp. 1–6, 2014.

[12] T. Mikolov, E. Grave, P. Bojanowski, C. Puhrsch, and A. Joulin, "Advances in pre-training distributed word representations," *Computation and Language*, 2017. arXiv:1712.09405.

[13] F. M. M. Mokbal, D. Wang, I. Azhar, J. Lin, F. Akhtar, and X. Wang, "MLPXSS: An integrated XSS-based attack detection scheme in web applications using multilayer perceptron technique," *IEEE Access*, vol. 7, no. 1, pp. 100567–100580, 2019.

[14] F. M. M. Mokbal, D. Wang, X. Wang, and L. Fu, "Data augmentation-based conditional wasserstein generative adversarial network-gradient penalty for

Table 5: Experiments results analysis for all models on a hold-out dataset

Model	Class	P	DR	F-S	AC	FN	FP	ER	ROC
LR	XSS	0.9809	0.9885	0.9847	0.9789	0.012	0.042	0.0211	9734
	Non-XSS	0.9746	0.9584	0.9664					
	macro avg	0.9777	0.9734	0.9755	(97.89%)	(1.2%)	(4.2%)	(2.11%)	(97.34%)
LDA	XSS	0.9806	0.9839	0.9822	0.9756	0.016	0.042	0.0244	0.9709
	Non-XSS	0.9649	0.9578	0.9613					
	macro avg	0.9727	0.9709	0.9718	(97.56%)	(1.6%)	(4.2%)	(2.44%)	(97.09%)
K-NN	XSS	0.9956	0.9856	0.9906	0.9871	0.014	0.009	0.0192	0.9881
	Non-XSS	0.9694	0.9906	0.9799					
	macro avg	0.9825	0.9881	0.9852	(98.71%)	(1.4%)	(0.9%)	(1.29%)	(98.81%)
CART	XSS	0.9748	0.9771	0.9760	0.9664	0.023	0.055	0.0336	0.9612
	Non-XSS	0.9503	0.9453	0.9478					
	macro avg	0.9625	0.9612	0.9619	(96.64%)	(2.3%)	(5.5%)	(3.36%)	(96.12%)
NB	XSS	0.9295	0.9673	0.9480	0.9274	0.033	0.159	0.0726	0.9043
	Non-XSS	0.9224	0.8412	0.8799					
	macro avg	0.9259	0.9043	0.9140	(92.74%)	(3.3%)	(15.9%)	(7.26%)	(90.43%)
AB	XSS	0.9759	0.9822	0.9790	0.9712	0.018	0.053	0.0288	0.9648
	Non-XSS	0.9609	0.9474	0.9541					
	macro avg	0.9684	0.9648	0.9666	(97.71%)	(1.8%)	(5.3%)	(2.88%)	(96.48%)
GBM	XSS	0.9842	0.9911	0.9877	0.9830	0.009	0.034	0.017	0.9784
	Non-XSS	0.9804	0.9656	0.9730					
	macro avg	0.9823	0.9784	0.9803	(98.30%)	(0.9%)	(3.4%)	(1.7%)	(97.84%)
RF	XSS	0.9904	0.9942	0.9923	0.9895	0.006	0.021	0.0105	0.9867
	Non-XSS	0.9874	0.9792	0.9833					
	macro avg	0.9889	0.9867	0.9878	(98.95%)	(0.6%)	(2.1%)	(1.05%)	(98.67%)
SVM	XSS	0.9954	0.9964	0.9959	0.9944	0.004	0.010	0.0056	0.9933
	Non-XSS	0.9922	0.9901	0.9911					
	macro avg	0.9938	0.9933	0.9935	(99.44%)	(0.4%)	(1.0%)	(0.56%)	(99.33%)

P= Precision, DR= Detection Rate, F-S= F-score, AC= Accuracy overall, FN= FN Rate, FP= FP Rate, ER= Misclassification rate (Error Rate), ROC= AUC-ROC

Table 6: Comparison with previous proposed works

Model	Accuracy	Precision	Detection Rate	F-score	FP	FN	ROC
Decision Tree [24]	-	0.9520	0.882	0.916	-	-	0.9479
CNN+SVM [1]	0.9937	0.9978	0.9886	0.9936	-	-	-
Random Forest [19]	0.972	0.977	0.971	0.974	0.087	-	-
NLP-SVM(This Work)	**0.9944**	**0.9954**	**0.9964**	**0.9959**	**0.010**	**0.004**	**0.9933**

xss attack detection system," *PeerJ Computer Science*, vol. 6, p. e328, 2020.

[15] E. Mugaboand and Q. Y. Zhang, "Intrusion detection method based on support vector machine and information gain for mobile cloud computing," *International Journal of Network Security*, vol. 22, no. 2, pp. 231–241, 2020.

[16] NVD.nist.gov. *NVD - vulnerability metrics*, 2020. (`https://nvd.nist.gov/vuln-metrics/cvss`)

[17] PortSwigger research. *Cross Site Scripting (XSS) Research*, 2020. (`https://nvd.nist.gov/vuln-metrics/cvss`)

[18] PreciseSecurity.com. *Cross-site scripting (XSS) makes nearly 40% of all cyber attacks in 2019*, 2020. (`https://www.precisesecurity.com/articles/cross-site-scripting-xss-makes-nearly-40-of-all-cyber-attacks-in-2019/`)

[19] S. Rathore, P. K. Sharma, and J. H. Park, "XSSclassifier: An efficient XSS attack detection approach based on machine learning classifier on SNSS," *Journal of Information Processing Systems*, vol. 13, no. 4, 2017.

[20] G. E. Rodríguez, J. G. Torres, P. Flores, and D. E. Benavides, "Cross-site scripting (XSS) attacks and mitigation: A survey," *Computer Networks*, vol. 166, pp. 106960, 2020.

[21] H. Shahriar and H. M. Haddad, "Client-side detection of clickjacking attacks," *International Journal of Information Security and Privacy (IJISP'15)*, vol. 9, no. 1, pp. 1–25, 2015.

[22] S. Shan, "Support vector machine," in *Machine Learning Models and Algorithms for Big Data Classification*, pp. 207–235, 2016.

[23] A. P. Sivanesan, A. Mathur, and A. Y. Javaid, "A google chromium browser extension for detecting XSS attack in HTML5 based websites," in *IEEE International Conference on Electro/Information Technology (EIT'18)*, pp. 0302–0304, 2018.

[24] R. Wang, Y. Zhu, J. Tan, and B. Zhou, "Detection of malicious web pages based on hybrid analysis," *Journal of Information Security and Applications*, vol. 35, pp. 68–74, 2017.

[25] Y. Wang, W. D. Cai, and P. C. Wei, "A deep learning approach for detecting malicious javascript code," *Security and Communication Networks*, vol. 9, no. 11, pp. 1520–1534, 2016.

[26] C. Wang and Y. Zhou, "A new cross-site scripting detection mechanism integrated with HTML5 and cors properties by using browser extensions," in *International Computer Symposium (ICS'16)*, pp. 264–269, 2016.

[27] H. Yulianton, H. Warnars, B. Soewito, F. L. Gaol, and E. Abdurachman, "Web security and vulnerability: A literature review," in *Journal of Physics: Conference Series*, vol. 1477, pp. 022028, 2020.

[28] I. Yusof, A. S. K. Pathan, "Preventing persistent cross-site scripting (XSS) attack by applying pattern filtering approach," in *The 5th International Conference on Information and Communication Technology for The Muslim World (ICT4M'14)*, pp. 1–6, 2014.

Biography

Fawaz Mokbal received his BS degree in Computer Science from Thamar University, Yemen, and MS degree in Information Technology from the University of Agriculture, Pakistan. He is currently a PhD researcher in Computer Science and Technology with Beijing University of Technology, China. He also an Assistant Professor with the Faculty of Computer Science at ILMA University, Pakistan. He has served as head of the Technical Team of Information Center Project for the local Authority for 2 years, and Manager of Information Systems at Ministry of Local Administration 5 years. He is the author and reviewer with various SCI, EI, and Scopus indexed journals. His interest area includes Machine and Deep Learning, Artificial Neural Networks, Medical Images, Web Application Security, and IoT security issues.

Wang Dan received the B.S. degree in computer application, the M.S. degree in computer software and theory, and the Ph.D. degree in computer software and theory from Northeastern University, China, in 1991, 1996, and 2002, respectively. She is currently a Professor with the College of Computer Science, Beijing University of Technology. She is the author and reviewer with various SCI, EI, and Scopus indexed journals. Her major areas of interests include trusted software, web security, and big data.

Wang Xiaoxireceived his MS degree in Computer Technology from Beijing University of Technology. He is currently working as an engineer in State Grid Management College. His major area of interest is Computer Network.

Research on Network Security Intrusion Detection with an Extreme Learning Machine Algorithm

Ying Xue

(Corresponding author: Ying Xue)

Shaanxi Police College, Xi'an, Shaanxi 710021, China

Email: yhn3br@163.com

(Received Apr. 13, 2020; Revised and Accepted May 25, 2021; First Online Nov. 9, 2021)

Abstract

With the rapid development of the Internet and the expansion of its application, network security is becoming more critical, especially the defense of network intrusion. This paper briefly introduced the network intrusion detection and extreme learning machine (ELM) algorithm, improved ELM with the kernel principal component analysis (KPCA) algorithm. Then, it carried out simulation experiments on the improved ELM algorithm. The simulation experiment included the parameter selection of the KPCA-ELM algorithm and the performance comparison between the support vector machine (SVM) algorithm and the ELM algorithm. The results showed that the optimal number of nodes in the hidden layer of the KPCA-ELM algorithm was 50, and the activation function was sigmoid. Furthermore, the KPCA-ELM algorithm had higher accuracy, lowered false alarm rate, and took less time than the other two detection algorithms.

Keywords: Extreme Learning Machine; Intrusion Detection; Kernel Principal Component Analysis; Network Security

1 Introduction

While the Internet brings convenience, it also brings corresponding risks. With the wide application of the Internet, users gradually upload important information to the network server, which often contains important privacy information. Once leaked, it will cause serious security risks [6]. The anonymity and connectivity of the Internet enable criminals to conduct more covert intrusion, steal or destroy important information, and cause losses to legitimate users [4]. The traditional network security protection measure to ensure network security is a firewall [5], *i.e.*, an isolation measure. Its principle is to set up a data monitoring system similar to the wall between multiple regional networks to shield attacks from the outside world as much as possible.

Network intrusion detection is an active network defense measure. The basic principle of network intrusion detection to achieve active defense is to detect the data actively with the detector connected with the network interface layer in the detection system. An intrusion detection system [9] can actively collect the data information of key nodes in the network and detect them to achieve real-time defense of network intrusion. Fossaceca *et al.* [3] proposed a new multiple adaptive reduction kernel extreme learning machine (ELM). The experimental results showed that it was scalable to large data sets. Eduardo *et al.* [2] put forward a network anomaly detection and classification method combining statistical technology with a self-organizing map, and the detection ability of the system could be modified without retraining the mapping. Park *et al.* [11] proposed a risk index quantifying the intrusion information through principal component analysis (PCA) to represent the comprehensive risk of port scanning attacks. The experimental results verified the excellent performance of the method.

2 Network Intrusion Detection

The detection methods of intrusion data in network intrusion detection systems are divided into misuse detection and anomaly detection [10]. Both methods need to learn to obtain the relevant data feature database and the parameters of detection algorithms. The difference between them is that misuse detection identifies the intrusion data or behavior with fixed features and form an intrusion data feature library by learning and collecting these characteristics, but the feature library constructed by anomaly detection after learning includes features of normal data and behaviors [12]. Misuse detection uses the feature library of abnormal data to test the data, which depends on the perfection of the feature library of abnormal data. Once the feature library is deficient, it cannot accurately test the type of abnormal data. Anomaly detection uses the feature library of normal data to test the

data. As long as the data does not conform to the feature library and exceeds some threshold, it is judged as abnormal data, and the normal data feature library is easier to be perfected than the abnormal data feature library [8].

3 ELM Algorithm

3.1 Introduction of ELM Algorithm

When the anomaly detection method compares the features of the detected data with the feature library, it will judge according to the similarity. The setting of the threshold is very important, but the diversity of normal data and the camouflage of abnormal data will lead to misjudgment. The diversity of data and the camouflage of abnormal data make the features used for recognition have rules, but the differences between different abnormal data and normal data are different, making the recognition rules of features have nonlinear properties.

Back-propagation (BP) neural network [15] can deal with the randomness and nonlinearity of data recognition features well. However, BP neural network needs a lot of historical data for training to ensure the accuracy of recognition, and multiple hidden layers will lead to the adjustment of many parameters during learning, which will affect the learning speed. Also, the data traffic in the network is large; thus, multiple hidden layers of the BP neural network will also reduce detection speed. The ELM algorithm is proposed to speed up learning and detection.

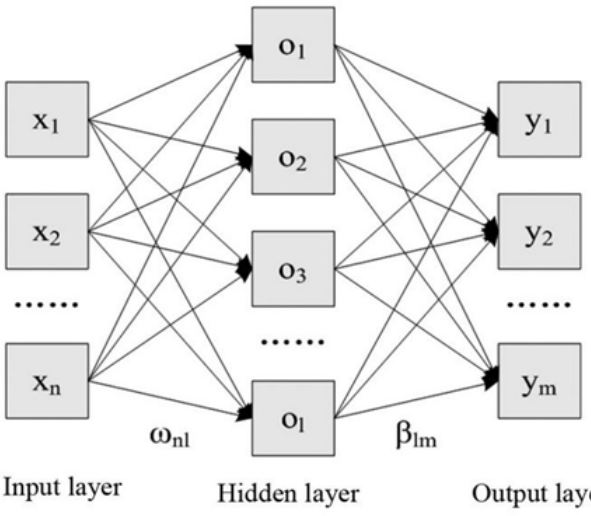


Figure 1: The basic structure of the ELM neural network

The ELM algorithm [1] is shown in Figure 1. The ELM algorithm also includes the input layer, hidden layer, and output layer, but there is only one hidden layer. The input layer has n neurons, corresponding to n input variables, *i.e.*, the feature dimension of the sample data in this paper. There are l neurons in the hidden layer, and its number is adjusted according to actual demands. There are m neurons in the output layer, corresponding to m output vectors.

BP neural network needs to reversely adjust its parameters step by step during training, which is a time-consuming and laborious process. The basic training principle of the ELM algorithm is to randomly generate and fix the weight ω between the input layer and the hidden layer and- the offset b between the three structural layers during calculation in ELM network, calculate the weight β between the hidden layer and output layer according to the training sample, equivalent to transform the training problem into a least square problem, and gradually adjust β to fit the data of the training sample on the premise of fixed ω and b. The basic steps of ELM training are as follows.

1) Firstly, the number of nodes in the hidden layer and the excitation function needed in the calculation are set.

2) Weight ω and offset b are randomly generated.

3) Weight value β is calculated using Equation (1) [13]: Where H is the output matrix of the hidden layer. H^+ is the generalized inverse matrix of H. T is the output matrix of the output layer. x_j is the j-th sample (totally Q samples). x_{n_j} is the n-th feature of x_j, and ω_i is the weight set of the i-th node in the hidden layer node to the nodes in the output layer.

4) After obtaining β, learning ends.

It is seen from the above steps that the ELM network, compared with the BP neural network, does not need to repeatedly adjust parameters such as weight and bias. The ELM network only needs to fit the training data under the fixed weight and bias of the hidden layer to obtain the output layer weight, and the training speed is higher.

3.2 Improving the Elm Algorithm with Kernel Principal Component Analysis

Although the ELM algorithm is fast in training, when it is applied to network intrusion detection, it takes much time to train and detect because of the large feature dimension of the data to be trained and detected, and not all the features have the same contribution to the data recognition. Therefore, to ensure the accuracy of recognition and improve the efficiency of training and detection, the ELM algorithm is improved by kernel principal component analysis (KPCA).

KPCA is a nonlinear extension of PCA [7], which is more suitable for analyzing the importance of nonlinear characteristics. The training flow of the improved ELM algorithm is shown in Figure 2, and the specific steps are as follows.

1) Training data are input, and a data matrix in a size of $m \times n$ (m is the number of training samples and n is

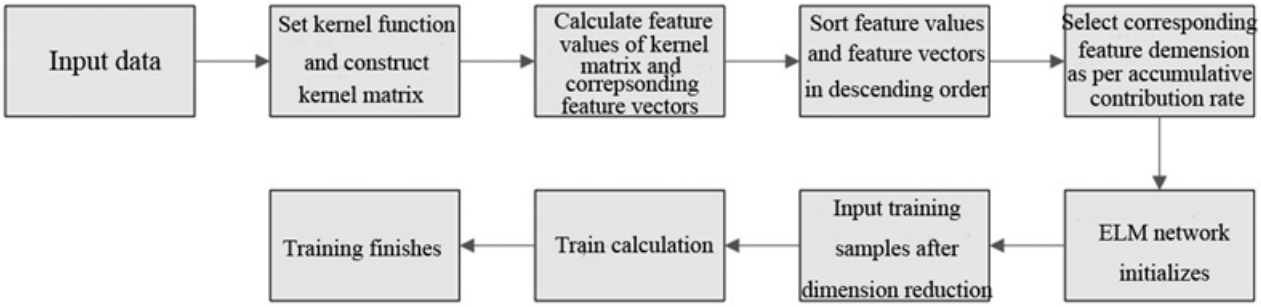


Figure 2: The ELM training process after improvement by KPCA

$$\begin{cases} \beta = (H^{+}H)H^{T}T \\ H = \begin{bmatrix} f(\omega_1 x_1 + b_1) & f(\omega_2 x_1 + b_{12}) & \cdots & f(\omega_1 x_1 + b_l) \\ f(\omega_1 x_2 + b_1) & f(\omega_2 x_2 + b_{22}) & \cdots & f(\omega_1 x_2 + b_l) \\ \vdots & \vdots & cdots & \vdots \\ f(\omega_1 x_Q + b_1) & f(\omega_2 x_Q + b_{n2}) & \cdots & f(\omega_1 x_Q + b_l) \end{bmatrix}_{Q\times l} \\ T = [t_1\ t_2\ \cdots\ t_Q]_{m\times Q} \\ t_j = \begin{bmatrix} t_{1_j} \\ t_{2_j} \\ \vdots \\ t_{m_j} \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^{l} \beta_{i_1} f(\omega_i x_j + b_i) \\ \sum_{i=1}^{l} \beta_{i_2} f(\omega_i x_j + b_i) \\ \cdots \\ \sum_{i=1}^{l} \beta_{i_m} f(\omega_i x_j + b_i) \end{bmatrix} (j = 1, 2, \cdots, Q) \\ \omega_i = [\omega_{1_i}, \omega_{2_i}, \cdots, \omega_{n_i}]\ (i = 1, 2, \cdots, l) \\ x_j = [x_{1_j}, x_{2_j}, \cdots, x_{n_j}]^T\ (j = 1, 2, \cdots, Q) \end{cases} \tag{1}$$

2) Kernel function $K(x_i, x)$ and its parameters are set, and a kernel matrix in a size of $m \times m$ is constructed. The elements in the kernel matrix are:

$$\begin{cases} K_{iv} = (\phi(x_i) \cdot \phi(x_v)) \\ i = 1, 2, \cdots, m \\ v = 1, 2, \cdots, m \end{cases} \tag{2}$$

where K_{iv} is an element in the i-th row and the v-th column in the kernel matrix, $\phi(x_i)$ and $\phi(x_v)$ are the mappings of samples x_i and x_v using kernel function $K(x_i, x)$.

3) The feature values and feature vectors of the kernel matrix are sorted in descending order.

4) The accumulative contribution rates of the first l sample features after descending sort are calculated. Features with an accumulative contribution rates exceeding the set range are selected as the main features.

5) The ELM network is initialized according to the traditional ELM algorithm steps, including randomly generating weight ω and offset b.

6) According to the dimension of principal component features given in Steps $1 \rightarrow 4$, the dimension of training samples is reduced, and then the samples are input into the ELM algorithm.

7) According to Equation (1), β in the ELM algorithm is calculated. The training ends.

When the improved ELM algorithm after training is used for intrusion detection, it also uses the principal component feature dimension processed by KPCA to reduce the dimension of the detected data before detecting the data with the trained ELM algorithm.

4 Simulation Experiment

4.1 Experimental Environment

The KPCA-ELM algorithm was simulated and analyzed by MATLAB software [14]. The experiment was carried out on a laboratory server, configured with Windows 7 system, I7 processor, and 16 G memory.

Table 1: Confusion matrix

	The algorithm is determined as abnormal	The algorithm is determined as normal
Abnormal actually	TP	FN
Normal actually	FP	TN

4.2 Experimental Data

The KDD99 data set was used in this study. Every data in the data set was 42-dimensional. The first 41 dimensions were the characteristic attributes of the data, and the last one was the decision attribute, indicating whether the data was abnormal or not, which was used for detecting the performance of the algorithm. One thousand normal data, 800 Dos data, 600 U2R data, and 400 Probe data were randomly selected from the KDD99 data set, and 30% of them were used as training samples and 70% as test samples.

4.3 Experimental Setup

Data set Preprocessing:
Among the 41 dimensions of features in the KDD99 dataset, only 38 dimensions of features were numbers, and the other three dimensions of features were characters, which could not be recognized directly by intrusion detection algorithms. Therefore, firstly, character features were converted into numerical features, and the 41-dimensional features became 122-dimensional numerical features. Then, the standardized operation was carried out to eliminate the problem of the large numerical span between different data:

$$y = \frac{x - \bar{x}}{x_{var}}, \tag{3}$$

where y is the standardized data; x is the data that needs to be standardized; $\bar{x}$ is the average value of the data; and x_{var} is the variance of the data.

The number of nodes in the hidden layer and the selection of activation function in the KPCA-ELM algorithm:
The KPCA-ELM algorithm is an improvement of the ELM algorithm. Except for reducing the dimension of the test data with the KPCA algorithm, the flow of other parts was consistent with the traditional ELM algorithm. The number of nodes in the hidden layer could be set freely, which would affect the training and detection, and the type of activation function in the hidden layer would also affect the accuracy of the detection algorithm. Therefore, in order to select the relatively optimal number of nodes in the hidden layer and the type of activation function, this paper compared the detection algorithms of three hidden layer activation functions, sigmoid, tanh, and relu. The number of nodes in the hidden layer of every active function was 20, 30, 40, 50, 60, 70, and 80, respectively.

Performance comparison between support vector machine (SVM), ELM, and KPCA-ELM algorithms:
The ELM algorithm is an algorithm before improvement, which was consistent with the ELM part of the KPCA-ELM algorithm. The relevant parameters of the three algorithms are as follows. The number of nodes in the hidden layer of the ELN algorithm was set as 50, and the activation function was sigmoid. The kernel function of the KPCA part in the KPCA-ELM algorithm was sigmoid, and the dimension with an accumulative contribution rate over 95% was taken as the reduced feature dimension. The kernel function of the SVM algorithm was sigmoid, and the penalty parameter was 1.

4.4 Performance Index

In this paper, the confusion matrix is used to analyze the detection performance of the three algorithms, and the accuracy and false alarm rate are used as the evaluation criteria (see Table 1):

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{4}$$

$$Error = \frac{FP}{FP + TN} \tag{5}$$

4.5 Experimental Results

When the KPCA-ELM algorithm detected intrusion data, its performance depends on the size of training samples and its parameter settings, including the number of hidden layer nodes and the activation function. This paper compared the algorithm performance under different numbers of nodes in the hidden layer and three activation functions to ensure the good performance of the KPCA-ELM algorithm in intrusion detection, and the results are shown in Figure 3. It was seen from Figure 3 that the accuracy of the detection algorithm under the three activation functions increased first and then decreased with the increase of the number of hidden layer nodes. When the hidden layer node was set as 50, the accuracy of the improved ELM algorithm under the three activation functions reached their respective highest values. By comparing the improved ELM algorithms that adopted different activation functions under the same number of hidden

Table 2: Time consumption between three detection algorithms for recognition

	SVM algorithm	ELM algorithm	KPCA-ELM algorithm
Dos/s	0.453	0.448	0.230
R2L/s	0.077	0.073	0.017
U2R/s	0.017	0.011	0.005
Probe/s	0.402	0.397	0.012
Integrated/s	0.237	0.232	0.066

layer nodes, it was found that the improved ELM algorithm that adopted the sigmoid function had a higher accuracy. Therefore, the KPCA-ELM algorithm finally chose the sigmoid function and set 50 hidden layer nodes as the final parameter setting for intrusion detection and subsequent performance comparison test.

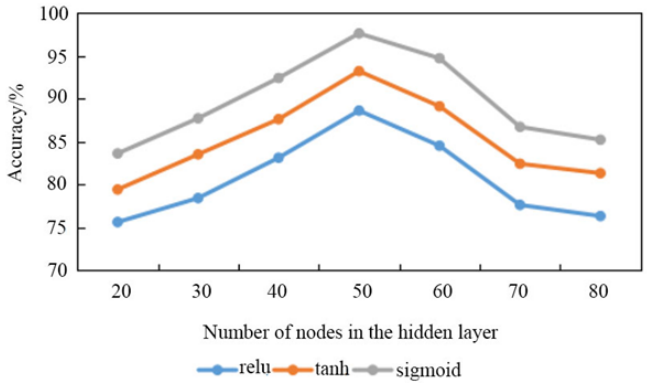


Figure 3: Accuracy of the KPCA-ELM algorithm under different activation functions and nodes in the hidden layer

The recognition accuracy of the three algorithms for intrusion data is shown in Figure 4. It was seen from Figure 4 that no matter what kind of intrusion data type, the KPCA-ELM algorithm always had the highest accuracy rate, the ELM algorithm was the second, and the SVM algorithm was the lowest. Also, when recognizing the integrated data, the KPCA-ELM algorithm had the highest recognition accuracy rate, and the SVM algorithm had the lowest accuracy rate.

The comparison of the false alarm rates between the three algorithms in Figure 5 showed that no matter what type of intrusion data, the SVM algorithm had the highest false alarm rate, followed by the ELM algorithm, and the KPCA-ELM algorithm had the lowest false alarm rate.

In conclusion, compared with SVM and traditional ELM algorithms, the KPCA-ELM algorithm had a higher recognition accuracy and lower false alarm rate. The reason for the above conclusion was that reducing the dimension of data with the KPCA algorithm reduced the amount of calculation, improved efficiency, and deleted the unimportant or even disturbing features among the data; in addition, the ELM algorithm was born out of the BP neural network, making it fit the nonlinear law of network intrusion data better than the SVM algorithm.

For network intrusion detection algorithms, in addition to the above detection accuracy and false alarm rate, the detection efficiency is also very important. If the efficiency of an intrusion detection algorithm is low, on the one hand, it can not find the intrusion data in time; on the other hand, the data can not pass before the detection is finished, thus the low detection efficiency will slow down the data transmission. The recognition time of the three detection algorithms is shown in Table 2. It was seen from Table 2 that in the face of different types of intrusion data, the three detection algorithms spent different time, but on the whole, the time consumption difference between the SVM algorithm and the ELM algorithm was not large, the ELM algorithm spent less time; the time consumption of the KPCA-ELM algorithm was significantly less than the other two algorithms, because the KPCA algorithm reduced the dimension of the detection data, greatly reducing the amount of calculation.

5 Conclusion

This paper briefly introduced network intrusion detection and the ELM algorithm used for network intrusion detection, and improved the ELM algorithm with the KPCA algorithm, and then carried out simulation experiments on the improved ELM algorithm. Through orthogonal experiments, the optimal number of hidden layer nodes and activation function of the KPCA-ELM algorithm were selected, and it was compared with SVM and traditional ELM algorithms. The results are as follows:

1) The orthogonal experiment showed that the optimal number of hidden layer nodes of the KPCA-ELM algorithm was 50, and the activation function was sigmoid;

2) For four types of intrusion data in the KDD99 dataset, the KPCA-ELM algorithm had the highest recognition accuracy, followed by traditional ELM and SVM algorithms;

3) For four types of intrusion data in the KDD99 dataset, the SVM algorithm had the highest false

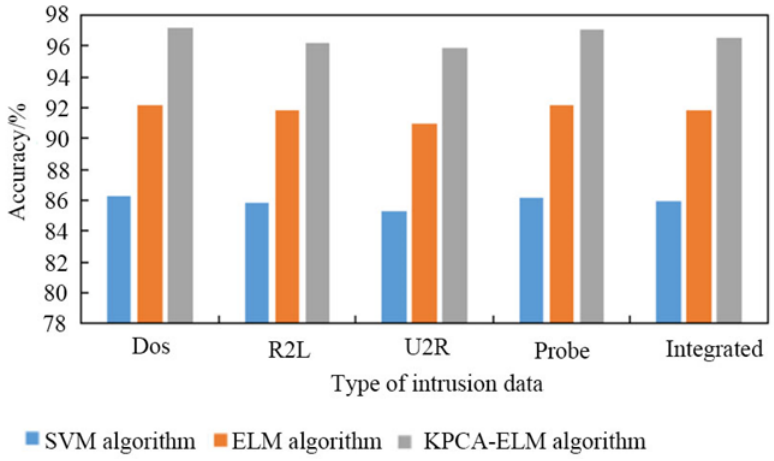


Figure 4: The recognition accuracy of three detection algorithms

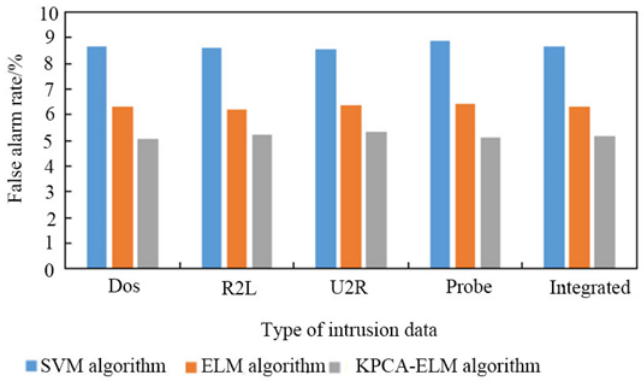


Figure 5: The false alarm rate of three detection algorithms

alarm rate, followed by the traditional ELM algorithm, and the KPCA-ELM algorithm had the lowest false alarm rate;

4) When recognizing intrusion data, the SVM algorithm took the longest time, followed by the ELM algorithm, and the KPCA-ELM algorithm took the shortest time.

References

[1] Z. Alom, V. R. Bontupalli, T. M. Taha, "Intrusion detection using deep belief network and extreme learning machine," *International Journal of Monitoring and Surveillance Technologies Research*, vol. 3, no. 2, pp. 35-56, 2016.

[2] H. E. Dela, A. Ortiz, J. Ortega, B. Prieto, "PCA filtering and probabilistic SOM for network intrusion detection," *Neurocomputing*, vol. 164, no. SEP.21, pp. 71-81, 2015.

[3] J. M. Fossaceca, T. A. Mazzuchi, S. Sarkani, "MARK-ELM: Application of a novel multiple kernel learning framework for improving the robustness of network intrusion detection," *Expert Systems with Applications*, vol. 42, no. 8, pp. 4062-4080, 2015.

[4] Y. Hamid, F. A. Shah, M. Sugumaran, "Wavelet neural network model for network intrusion detection system," *International Journal of Information Technology*, vol. 11, pp. 251-263, 2019.

[5] E. Hodo, X. Bellekens, A. W. Hamilton, P. L. Dubouilh, E. Iorkyase, C. Tachtatzis, R. C. Atkinson, "Threat analysis of IoT networks using artificial neural network intrusion detection system," *Tetrahedron Letters*, vol. 42, no. 39, pp. 6865-6867, 2017.

[6] N. Keegan, S. Y. Ji, A. Chaudhary, C. Concolato, B. Yu, D. H. Jeong, "A survey of cloud-based network intrusion detection analysis," *Human-centric Computing and Information Sciences*, vol. 6, no. 1, pp. 19, 2016.

[7] M. A. Khan, A. R. Sakhawat, K. M. Khan, M. A. A. Ghamdi, S. Almotiri, "Enhance intrusion detection in computer networks based on deep extreme learning machine," *Computers, Materials and Continua*, vol. 66, no. 1, pp. 467-480, 2020.

[8] Y. C. Li, R. X. Qiu, S. T. Jing, "Intrusion detection system using online sequence extreme learning machine (OS-ELM) in advanced metering infrastructure of smart grid," *PloS One*, vol. 13, no. 2, pp. e0192216, 2018.

[9] T. Ma, F. Wang, J. Cheng, Y. Yu, X. Y. Chen, "A hybrid spectral clustering and deep neural network ensemble algorithm for intrusion detection in sensor networks," *Sensors*, vol. 16, no. 10, pp. 1701, 2016.

[10] A. Q. Majjed, L. Yu, A. Mohammed, A. S. Kamal, "Deep learning approach combining sparse Autoencoder with SVM for network intrusion detection," *IEEE Access*, vol. 6, pp. 52843 - 52856, 2018.

[11] S. Park, J. Kim, "A study on risk index to analyze the impact of port scan and to detect slow port scan

in network intrusion detection," *Advanced Science Letters*, vol. 23, no. 10, pp. 10329-10336, 2017.

[12] R. Singh, H. Kumar, R. K. Singla, "An intrusion detection system using network traffic profiling and online sequential extreme learning machine," *Expert Systems with Applications*, vol. 42, no. 22, pp. 8609-8624, 2015.

[13] S. Sulandri, A. Basuki, F. A. Bachtiar, "Metode deteksi intrusi menggunakan algoritme extreme learning machine dengan correlation-based feature selection," *Journal Teknologi Informasi dan Ilmu Komputer*, vol. 8, no. 1, pp. 103, 2021.

[14] Y. Wang, A. N. Wang, Q. Ai, H. J. Sun, "An adaptive kernel-based weighted extreme learning machine approach for effective detection of Parkinson's disease," *Biomedical Signal Processing and Control*, vol. 38, pp. 400-410, 2017.

[15] Y. Yu, Z. Ye, X. Zheng, C. Rong, "An efficient cascaded method for network intrusion detection based on extreme learning machines," *Journal of Supercomputing*, vol. 74, no. 11, pp. 5797-5812, 2018.

Biography

Ying Xue, born in 1979, has received the master's degree from xi'an jiaotong University in 2007. She is working in Shaanxi Police College now as a associate professor. She is interested in network security.

Research on Monitoring Technology of Industrial Cannabis Based on Blockchain and SM Series Cryptographic Algorithm

Zijian Ma[1], Zhiqiang Wang[1,2], Hang Wu[3], Xingyu Guo[2], and Xizhen Wang[3]
(Corresponding author: Zhiqiang Wang)
School of Computer and Information Technology, Beijing Jiaotong University[1]
Beijing 100044
Email: wangzq@besti.edu.cn
Beijing Electronic Science and Technology Institute[2]
School of Telecommunications Engineering, Xidian University[3]
(Received Dec. 16, 2020; Revised and Accepted June 6, 2021; First Online Nov. 9, 2021)

Abstract

In China, the production and marketing relationship of industrial cannabis is becoming more and more complex. The existing supervise means are mainly manual, with low efficiency and security. Therefore, this paper proposes a traceable supply chain monitoring solution based on the private chain and proof of work (PoW). The records of our scheme including the source, the destination, the amount, and the loss of industrial cannabis for each transaction. The directed graph is used to abstract the transaction process. And three types of illegal behavior can be identified by alarm strategies. SM3 Cryptographic Hash Algorithm is used for integrity protection, and SM9 Identity-Based Cryptographic Algorithm is used for authenticity and confidentiality protection. In the experiment, with the verification system designed, the PoW calculation is stable, and the optimized encryption algorithms work efficiently.

Keywords: Blockchain; Industrial Cannabis; SM3; SM9

1 Introduction

Cannabis represents one of the most important global growth opportunities in the next decade. In China, after Yunnan and Heilongjiang, Jilin will become the third province to liberalize the cultivation of industrial hemp. During the cannabidiol (CBD) extraction and production, it is possible to purify THC as the raw material of drugs. Therefore, the entire production process must be strictly controlled by the public security department, with all aspects of the network monitoring. With the increasingly complex relationship between the production and marketing of industrial hemp, problems of low efficiency and low safety are exposed. It is urgent for public security departments to trace industrial hemp and its related products of information, and effectively supervise the production, processing, circulation, and sales of industrial hemp.

This paper studies the monitoring technology of industrial hemp based on the blockchain, and adopts the Chinese cryptographic algorithm to realize the security and reliability of the monitoring technology, to improve the regulatory efficiency of industrial hemp and the security of the monitoring information.

2 Related Work

Due to its high economic value, China's industrial hemp cultivation scale is constantly expanding, and its products are also applied in various fields [4]. Fiber and seed are the main products of industrial hemp. The former is widely used as textile raw materials due to its breathable and bacteriostatic properties. The latter has edible and medicinal value. Hemp materials, such as flowers and leaves, are also of value after processing, from which CBD can be extracted and used in the production of nutrition products, skin-care products, and energy drinks.

2.1 THC and Related Purification Techniques

Generally, the mass fraction of THC in industrial hemp is less than 0.3 % and has no drug utilization value. Human ingests hemp containing more than $3.0 gkg^{-1}$ of THC (expressed as mass concentration), then it shows that a certain tendency of medicinal or abuse [16]. At the present stage, mature technologies such as chromatographic production [3] and supercritical fluid [14] can be used to efficiently extract and obtain high-purity THC from industrial hemp flowers, stems, and leaves. This means that it is possible to purify drugs in the whole process of in-

dustrial hemp production and processing, which indicates that risks exist in each part of the industrial hemp supply chain.

2.2 Industrial Hemp Supply Chain Analysis Method

Existing analysis methods for the cannabis supply chain include Predator-Prey based model [5], Classification Trees [19], Information Fusion Predictive model based on medical approaches [1], and Embedding role Classification based on Compositional Multiview [2]. Among them, the Predator-Prey based model analyzes illicit drug consumption by its users. Classification Trees quantifies and predicts the use of industrial hemp products. Information Fusion Predictive model presents a supply and demand analysis method based on online social network data. Classification of the role based on Compositional Multiview Embedding dynamically feedback relevant information of individual users, informed organizations, and retail accounts by integrating time and geographical location. However, with the increasingly complex relationship between its production and marketing, the above methods cannot meet the need for comprehensive and traceable monitoring.

2.3 Blockchain and Proof-of-Work

Although the specific implementation of blockchain technology is different, it still has many generalities in the whole system architecture. Taking Bitcoin and Ethereum as examples to analyze the general structure of blockchain, can be divided into five layers shown in Figure 1, which are the application layer, thecontrol layer, the consensus layer, the network layer, and the data layer. Among them, the control layer includes automatic scripts and sandbox environment written by the program, and the data layer includes block data structure and specific storage content.

Due to its characteristics of openness, cooperation and mutual trust, decentralization, and confidentiality, and reliability, blockchain technology has been applied to information sharing and product traceability, with most application scenarios in finance and medical treatment. Typically, blockchain can realize distributed Shared ledger and generate intelligent financial reports [7], realize fair payment contracts in cloud storage public audit [13], solve electronic medical record management problems in Internet of Things devices [12], and build a reverse supply chain of expired drugs [18]. Trust relationship and privacy protection are the key points in the practical application of blockchain, which need to be supported by reliable algorithms. Existing blockchain consensus algorithms include Proof of Work (PoW), Proof of Stake(PoS), Practical Byzantine Fault Tolerance (PBFT) and Raft algorithm, *etc.* [11]. Under the further study, Chinese researchers proposed a PBFT [20] based on a tree-topology network and a Raft [8] combined with a BLS signature. However, the blockchain hash algorithm is dominated by SHA-256 [15], and the asymmetric encryption algorithm is dominated by RSA, which is relatively limited and has some security risks, so it is not conducive to domestic generalize. Beijing Electronic Science and Technology Institute has proposed a provable security block chain privacy protection scheme based on the SM9 algorithm [17], but it only focuses on group signature and is limited to theoretical analysis, without combining specific application scenarios.

3 The Design of Blockchain

Combined with the Chinese cryptographic algorithm and the actual application requirements of industrial hemp monitoring, this paper focuses on the blockchain data layer. The storage model based on account stores data in the form of key-value pairs and constantly updates the account data by executing transactions, which is not convenient for the comprehensive and specific recording of industrial hemp flow information. In order to realize the reverse traceability of the supply chain, a private chain was constructed based on the transaction and the circulation of each batch of industrial hemp was recorded.

3.1 Proof of Work

This paper designs a private chain based on the Proof of Work (PoW) algorithm and controls the block generation. The node matches the target by calculating, and then generates the block.

The design of PoW is not complex, and the computational overhead is relatively stable. The reasons are as follows. First, private chain nodes are dominated by personal computers and do not have great computing power. Second, PoW sets the number of unrecorded transactions as a trigger condition. And the Stable computational overhead means a stable time from PoW to a new block generation. The number of transactions generated is limited so that the number of transactions recorded in each block is similar. Third, the purpose of the new block is generated for recording transactions. Nodes are not rewarded for finding blocks, and it is meaningless to skip the PoW attack strategy.

Starting from the work's initial value, the calulation process of each iteration is shown in Table 1. Current proof p' is increment by the previous proof p, and k is the increment. Before each increment, p and p' are concatenated as a string, and the result is substituted into the SM3 hash algorithm. If the number of 0 at the end of the hash is l, then hit the target, otherwise continue to calculate. In the verification system, the initial value of work 1, and k=3,l=1, the reasons are shown in Section 6.1.

3.2 Block Structure

The creation block and the general block are designed as the basic components of the block chain. This section de-

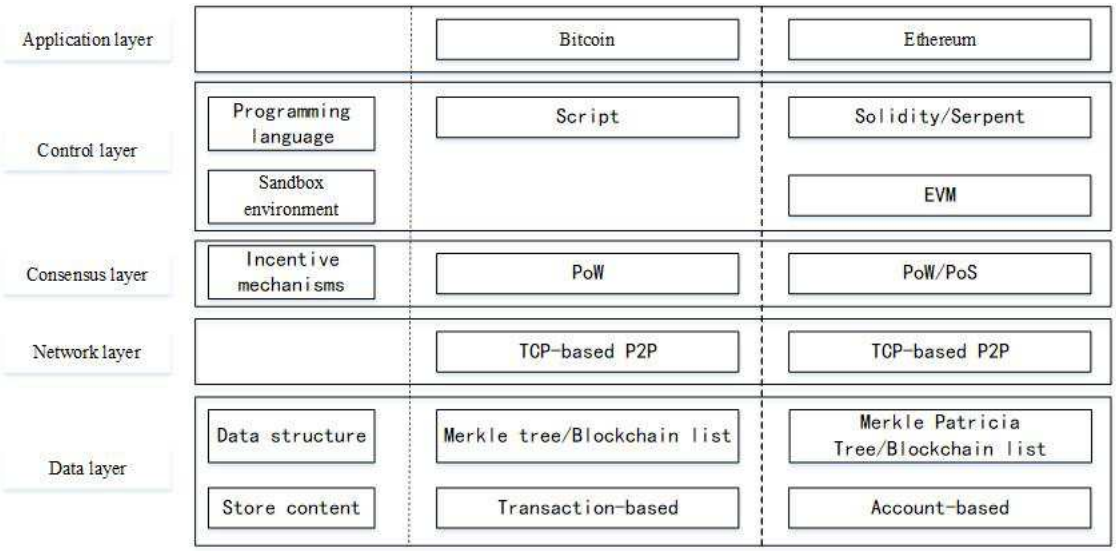


Figure 1: The proposed scheme

Table 1: Calculation process of PoW

Input: Pervious proof p Output: Current proof p'
Begin: $p' = p$ While $l! = 0$: $p' = p' + k$ End while End

scribes the two block structures respectively, as shown in Figure 2. In the header of the block, the previous block hash (PreBlockHash) is calculated by the SM3 hash algorithm, the Proof is calculated by the POW algorithm, and the block serial number (Index) and Time stamp (Time) are automatically generated.

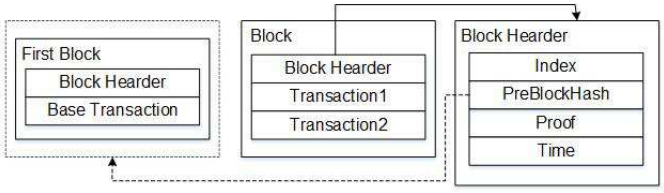


Figure 2: Industrial hemp block structure

The creation block is the first block in the private chain and can only be initiated by industrial hemp producers. And it only stores the corresponding industrial hemp production batches of Base Transaction. There is no preblock, so the hash of the previous block is the constant value 100, and the work is set to the initial value of 1.

Generally, a block has and only has one precursor, and each node is generated by work calculation. The record involves multiple transactions of this batch of industrial hemp. There are three reasons to eliminate incentives. First, the blockchain used is private and does not require or attract other computing power to participate in the PoW. Second, non-profit, but for recording transactions. Third, from a security perspective, it makes no sense to attack incentives such as Selfish Mining.

3.3 Transaction Structure

Transactions initiated by each node are recorded in the block in chronological order. In the context of industrial hemp application, the transaction can correspond to a certain process, as well as market behaviors such as transportation and purchase. In particular, the initial transaction refers specifically to the first transaction recorded in the creation block, the property of which is described in Section 2.4.

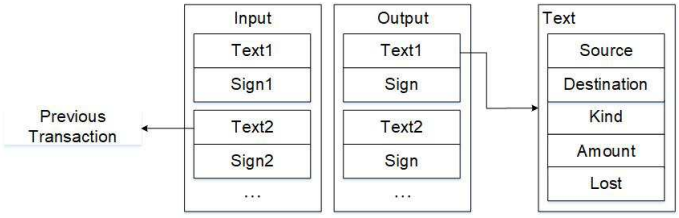


Figure 3: Transaction structure of industrial hemp

The designed transaction structure is shown in Figure 3.,where the target of the Sign is the Source of the sender. The Kind of industrial hemp product is transmitted in ciphertext, and only the Destination can be seen after decryption. Normal loss may be caused during processing and transportation, so in order to comprehensively and concretely reflect and quantify the flow of industrial hemp products, we record Amount and Lost by quality.

3.4 Monitoring and Alarm Policies

The transaction behavior in the block chain is abstracted by the directed generation graph D=(V,E,C). Among them,$\mathbf{V} = \{v_0, v_1, v_2...\}$ is the node set, v_0 is the super source point, v_1 is the originating node of the creation block. E is a directed edge set, and $e_{ab} \in E$ represents node v_a to send a transaction to the node v_b. C is the directed edge capacity set, c_{ab}=(i,j),$c_{ab} \in C$ represents the transaction in e_{ab}, the industrial hemp transaction is I and the loss is J.

The input degree of super source node v_0 is 0, and the output degree is 1, the corresponding entity is the same as v_1 . Industrial hemp producers initiate creation blocks that may send the initial trade directly to specific nodes $v_0 \to v_1 \to v_a$. It was also possible to send an initial transaction to myself and then initiate a transaction to multiple nodes$v_0 \to v_1 \to W$, $W \subseteq V$. Whatever in any case, $c_{01} = (i, j)$,j=0, i is always the sum of the quality of the industrial hemp batch. In the former case, v_a obtained all the industrial hemp in that batch. In the latter case, v_1 distributes the industrial hemp batch to multiple nodes. Blockchain records all transactions involving the batch of industrial hemp means $c_{01}[0] = \sum c_{ab}[1] + \sum c_{ba}[0] - \sum c_{ab}[0], a \neq 0, b \neq 0$, and the sum of the total loss and the amount of each node is equal to the initial trades.

When $a \neq 0$, a new transaction e_{ab}, is initiated, an exception is determined, and the transaction is terminated immediately. First, the receiver doesn't exist, that is $v_b \notin V$. Second, the node $v_a$lack holdings, that is $\sum c_{ka}[0] - \sum c_{ak}[0] - \sum c_{ak}[1] < i + j$,$c_{ab}$=(i,j). Third, the quality loss is over expected, that is $j \geq n * i$,c_{ab}=(i,j), n is the proportionality constant, which can be adjusted according to the actual situation. And n is equal to 1 in the verification system.

4 SM3 Hash Cryptographic Algorithm

SM3 hash algorithm was released as the national standard by State Cryptography Administration in 2016. Merkle-Damgard structure is adopted in the algorithm. And for input messages of variable length, 256 bit hash value is generated and output after filling and iterative compression [10]. This section describes the software optimization implementation of SM3 used in the blockchain.

4.1 Constants and Functions

Initial value IV consists of eight 32-bit series:

$$\begin{aligned} \text{IV} = \quad & 7380166\text{f} \;\; 4914\text{b}2\text{b}9 \;\; 172442\text{d}7 \;\; \text{da}8\text{a}0600 \\ & \text{a}96\text{f}30\text{bc} \;\; 163138\text{aa} \;\; \text{e}38\text{dee}4\text{d} \;\; \text{b}0\text{fb}0\text{e}4\text{e} \end{aligned}$$

Constant T_j is used in modular 232 addition operation to reduce the linear and differential genetic probability between input and output [6]:

$$T_j = \begin{cases} 79cc4519, & 0 \leq j \leq 15 \\ 7a879d8a, & 16 \leq j \leq 63 \end{cases}$$

Boolean function, $\wedge$ represents 32-bit **AND** operation,$\vee$ is 32-bit **OR** operation, $\oplus$is 32-bit **XOR** operation,$\neg$is 32-bit **NOT** operation:

$$FF_j(X,Y,Z) = \begin{cases} X \oplus Y \oplus Z, & 0 \leq j \leq 15 \\ (X \wedge Y) \vee (X \wedge Z) \vee (Y \wedge Z), & 16 \leq j \leq 63 \end{cases}$$

$$GG_j(X,Y,Z) = \begin{cases} X \oplus Y \oplus Z, & 0 \leq j \leq 15 \\ (X \wedge Y) \vee (\neg X \wedge Z), & 16 \leq j \leq 63 \end{cases}$$

Replacement function, $\oplus k$ represents k-bit ring left shift of 32-bit.

$$\begin{aligned} P_0(X) &= X \oplus (X <<< 9) \oplus (X <<< 17) \\ P_1(X) &= X \oplus (X <<< 15) \oplus (X <<< 23). \end{aligned}$$

Calculate 32 bit ring left shift K bit, $X \gg (32-k) || X \ll k$. Under the condition of multi-digit abandonment, then take the **XOR** operation between the result of moving X to the left k bit and X to the right 32-k bit.

4.2 Computational Flow and Software Optimization

We will fill the message m whose length L is not a multiple of 512. We add 1 bit "1", k bit "0", and a 64-bit string at the back of message m. Where K is the minimum non-negative integer satisfying l+k+1=448 **mod** 512, and the 64-bit string is the binary representation of length L. Take message 00000000 as an example, its length L =8, and the filling process is shown in Figure 4.

Figure 4: Example of SM3 message padding

Optimize the process with the help of bytes structure and mathematical reasoning.Encode the message m and get the bytes array, and convert it to a hexadecimal string. Take four bits as a group, and the filling rules are as follows. The first 1-bit is "1" which is combined with the subsequent 3-bit "0", then add them to hexadecimal string as an "8". The remaining k-3 bit "0" combinations are added as hexadecimal "0" and the number is (k-3) // 4. For the 64-bit strings, calculate the hexadecimal length l for h (leading 0 is reserved) firstly. Add hexadecimal "0" with the number of 16-L_h, where L_h is the hexadecimal length of H. Finally, add h. After coding, l % 4=0, and because of k=447-l **mod** 512, so $k > 3$ and $(k-3)\%4 \equiv 0$ when $0 < l < 512$. So we can optimize by bytes array, and the time complexity is reduced from O (k) to O (k/4).

Table 2: Iterative compression process

Input: message grouping $B^{(i)}$, initial value IV Output: compression results $V^{(n)}$

for $i = 0$ *to* $n-1$ do
$V^{(i+1)} = CF(V^{(i)}, B^{(i)})$
end for

Table 3: Message extension process

Input: Divide $B^{(i)}$ to get 16 words $W_0, W_1 ... W_{15}$ Output: $W_{16}, W_{17} ... W_{67}, W_0', W_1' ... W_{63}'$

for $j = 16$ *to* 67
$W_j = P_1(W_{j-16} \oplus W_{j-9} \oplus (W_{j-3} <<< 15)) \oplus (W_{j-1} <<< 7) \oplus W_{j-6}$
end for
for $j = 16$ *to* 63 do
$W_j' = W_j \oplus W_{j+4}$
end for

Iteratively compress the filled message m′. Then the message m′ is grouped with a length of 512 bits, denoted as B^i, where $0 \le i \le (l+k+65)/512 - 1$. The iterative process is shown in Table 2, where CF is the compression function, V^0 is the initial value IV, V^n is the compression result, and n=(l+k+65)/512. Message expansion in the unit of message grouping B^i, 132 message words are generated for the compression function, denoted as W_0, W_1 $\cdots$, W_{67}, W_0', W_1', $\cdots$, W_{63}', it is shown in Table 3.

In big-end storage mode, CF calculation process is shown in Table 4. Where *A,B,C,D,E,F,G,H* are directly calculated as integer data, and SS1, SS2, TT1, TT2 are intermediate variables.

5 SM9 Identification Cryptographic Algorithm

SM9 identity cipher algorithm is a bilinear pair-based identity cipher algorithm, published as the national standard by the State Cryptography Administration in 2015. It generates public and private key pairs based on the user's identity, and the identity information is associated with the password algorithm, eliminating the process of digital certificate, certificate library, and keystore management. This section introduces the specific functions of the algorithm and the implementation of software optimization.

5.1 Propaedeutics

The mathematical tools of SM9 are point group operation and bilinear pair operation of elliptic curve over finite field group. Let G_1 and G_2 be two additive cyclic groups of order N, G_T be a multiplicative cyclic group of order N, N is a prime number, and $N > 2^{191}$. P_1 and P_2 are generators of G_1 and G_2 respectively.

Point group operation is the basis, it includes addition and multiple points. The addition operation is the mod N addition of G_1 and G_2 domain and the symbol is expressed as m+n=(m+n) *mod* N, where $m, n \in G$. The multiple point operation is the repeated addition of the same point on an elliptic curve.

Bilinear pair operation e(G_1, G_2) is mapping $G_1 \times G_2 \to G_T$. Four kinds of intractable problems are used to ensure the safety of the algorithm.

Problem 1, bilinear inverse (BIDH), given ($a[P_1]$, $b[P_2]$), $a, b \in [1, N-1]$, to calculate $e(P_1, P_2)^{b/a}$ difficult.

Problem 2, judgmental bilinear inverse (DBIDH), given $(a, b, r) \in [1, N-1]$, it is difficult to distinguish between (P_1, P_2, $[a]P_1$, $[b]P_2$, $e(P_1, P_2)^{b/a}$).

Problem 3, τ-bilinear inverse (τ-Gap-BDHI), given (P_1, $[x]P_1, P_2, [x]P_2, [x^2]P_2, \cdots, [x^\tau]P_2), \tau, x \in [1, N-1]$ computing $e(P_1, P_2)^{1/x}$ difficult.

Problem 4, τ-Gap-bilinear inverse (τ-Gap-BDHI), given (P_1, $[x]P_1, P_2, [x]P_2, [x^2]P_2, \cdots, [x^\tau]P_2), \tau, x \in [1, N-1]$ computing $e(P_1, P_2)^{1/x}$.

5.2 The General Process

According to functions, four modules can be divided: parameter initialization, calculation initialization, key derivation, and password calculation. The calculation process is shown in Figure 5.

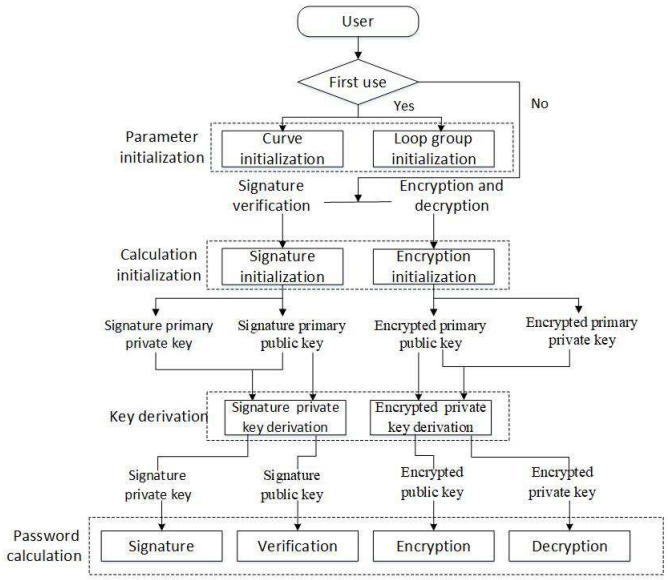


Figure 5: SM9 function module and calculation process

The parameter initialization module provides calculation parameters. The curve initialization completes the definition of finite field and elliptic curve. The loop group

Table 4: The compression function of the calculation process

Input: $W_0, W_1 ... W_{67}, W_0', W_1' ... W_{63}', V^{(i)}$ Output: $V^{(i+1)}$

$ABCDEFGH \leftarrow V^{(i)}$
for $j = 0$ *to* 63 do
$SS1 \leftarrow ((A <<< 12) + E + (T_j <<< (j \mod 32))) <<< 7$
$SS2 \leftarrow SS1 \oplus (A <<< 12)$
$TT1 \leftarrow FF_j(A,B,C) + D + SS2 + W_j'$
$TT2 \leftarrow GG_j(E,F,G) + H + SS1 + W_j$
$D \leftarrow C$
$C \leftarrow B <<< 9$
$B \leftarrow A$
$A \leftarrow TT1$
$H \leftarrow G$
$G \leftarrow F <<< 19$
$F \leftarrow E$
$E \leftarrow p_0(TT2)$
end for
$V^{(i+1)} = ABCDEFGH \oplus V^{(i)}$

initializes the calculated generators P_1 and P_2. For each user, generate tuples that are unique and generally not modified. So parameter initialization is only done for the first use by users, or when information security is compromised and the generation tuple must be regenerated.

The calculation initialization module realizes the generation of the primary key. In signature initialization, the signature's master private key s is generated randomly. The signature's primary public key S is s times P_2, published with a bilinear pair of P_2 and S. In the encryption initialization, the encryption main private key e is generated randomly. The encrypted primary public key E is e times P_1, published with a bilinear pair of E and P_2.

The key derivation module uses SM3 as a auxiliary hash function to derive the key based on the primary key and user identity. When the signature private key is derived, the user identity participates in the hash operation, s is added to the hash value, and after modulus inverse, the result is multiplied with P_1 to generate the signature private key. When the signature private key is derived, the user identity participates in the hash operation, and the sum of e and the hash value is performed. After modulus inverse, the result is multifold with P_2, and finally, the encrypted private key is generated.

The cipher calculation module provides four functions: signature, check, encryption and decryption. When signing, the generated signature private key is used. To verify the signature, SM3 is used to generate a hash value of the user id. The result is multipoint calculated with P_2, and then added with the signature's primary public key. The final result is used to verify the signature. To encrypt it, SM3 is used to generate a hash of the user id. The result is multiplied by P_1, then added to the encrypted primary public key and encrypted with the final result. When decrypted, the generated encrypted private key is used.

5.3 Software Optimization

Key derivation function *KDF (Z,L)* is called in key derivation module, cipher calculation module, and other relevant modules calculation. Its input Z is the data to be processed and L is the length of data to be obtained. In this paper, SM3 is used to assist KDF calculation. Consistent with the output format of SM3, KDF computations no longer use bit strings, but are grouped into hexadecimal strings. Z is entered directly as a hexadecimal string, and L indicates the desired hexadecimal string length. The 32-bit registers involved in the computation are represented as 8-bit hexadecimal in the implementation of the software. Compared with the computation bit by bit, the time complexity of the algorithm is reduced by 1/4.

Except for KDF, other data information processed by key derived module and cipher calculation module is expressed by hexadecimal. Hexadecimal strings have an advantage over strings in computation involving a decimal integers. As shown in Figure 6, if "4d " appears as a character, the conversion to a decimal integer must be encoded firstly, then each character is represented as a decimal integer and finally stored as a list. If "4d " is regarded as hexadecimal, it can be converted directly into hexadecimal, with less computational overhead. In particular, it is different from the hexadecimal representation of Large Numbers, we use bytes as a structure for grouping, which takes up less space. Take "test" as an example, as the information to be processed, its hexadecimal representation is e6b58be8af95, with "b" data storage, occupying 39 memory. It is stored with type 0x data, occupying 69 memory.

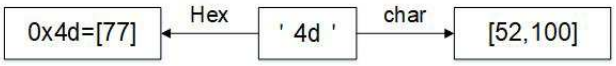


Figure 6: Results of type conversion under different strategies

6 Verification System Design

The verification system is based on B/S architecture and realizes application layer interaction through Web pages. Through this system, the block running information is checked, and the block running condition is detected.The system provides a verification platform for SM3 and SM9 related experiments. This section introduces its function module design, running process, and boundary treatment method.

6.1 Development Environment

Javascript, JS for short, is a widely used literal translation scripting language. In addition to UI (User Interface) design, the verification system uses this language for access control and boundary handling. With the help of this language, the validity of the input data on the web page can be verified, and the communication data encapsulation with server and communication result feedback can be realized in the verification system.

JSON (Javascript Object Notation) is a lightweight data exchange format. In Python, this format can be solved by flask library functions. It is simple and flexible. Blockchain uses this format to record block and transaction information. In the verification system, the communication data transmitted between the server and the browser is mainly in this format.

AJAX (Asynchronous JavaScript And XML) is for interactive web pages that update web content without reloading. In the verification system, the web page uses this technology to interact with the server, request data in the process of bill backtracking, holding query and block verification, and display the data on the page. Submit data when the new transaction is initiated and the block created, receive and display feedback information from the server.

6.2 Function Module and Running Process

The verification system is designed to meet both experimental and application requirements. It can be divided into three modules according to functions: data interaction module, transaction initiation module, and query detection module, as shown in Figure 7. The data interaction module authenticates the user. The interface includes a user identity submission interface and an identity file upload interface. The user id is submitted for the first time if the identity file has not been generated. The user id is used to generate an identity file, after the verification system receives it, SM9 is called for operation, and the calculated results such as a generator, primary public key, and primary private key are encapsulated and returned in the form of the file. For users with existing identity files, the verification system will verify the files, as detailed in Section 5.3. Allow the user to log in after confirming the identity file is correct.

Figure 7: Interface of verify system and function module

The transaction initiation module collects the specific information needed by the creation block and the new transaction initiation. On the interface, a text box prompts you to enter the "recipient" of industrial hemp trade, the "type" of industrial hemp traded, the "amount" of industrial hemp involved in the trade, and the normal "loss" of industrial hemp caused by the current node. The "quantity" and "loss" of industrial hemp were expressed in terms of quality (gram). Specifically, the industrial marijuana trade "sender" is fixed for the identity file read by the user id. The user clicks the "Start" button, and the input information is submitted as the initial transaction. The corresponding creation block is established, and the initial transaction is directly written into it. When the user clicks the "trade" button, the input information will be submitted as general trade, and the writing location will be determined according to the actual situation. Boundary handling and illegal input filtering are shown in Section 5.3.

The query and detection module provides three functions: "backtracking bill", "holding query" and "block check". Among them, "backtracking bill" returns all the current blockchain information, so that users can visually view the block generation and transaction records. Among the returned information, "signature" is the calculation result of SM9 signature algorithm, and the "sender" signature private key participates in the operation. "Type" is the result of SM9 encryption algorithm calculation, and "receiver" encrypted public key participates in the calculation. The "Possession query" returns the quality of the current user's industrial hemp possession, that is, the difference between the total input and output qualities. "Block check" returns the verified blockchain information, including workload proof, hash of the previous block, and user signature. Among them, if the signature is approved, the text "true" is returned. If the signature verification fails, the text "false" is returned. In particular, to confirm the correctness of encryption and decryption calculations, "category" is returned as SM9 decryption results, in contrast to the "back billing" return information.

In general, a new block is created using the verification system to record transactions, which can be done as shown in Figure 8. First, enter the user id in the data interaction module to get the identity file. The existing identity file is uploaded and read correctly in the data interaction module before entering the main interface. For certain batches of industrial hemp, the creation block cannot be initiated for general trade until it is produced. User A and user B log in respectively, and industrial marijuana consumption is 0 by default. User A, as an industrial hemp producer, initiated the initial transaction and generated the creation block. As the receiving party, user B's industrial marijuana consumption increases after receiving the transaction. At this point, User B can initiate a general transaction, but the transaction volume is limited and the sum of "quantity" and "loss" cannot exceed the total amount of industrial marijuana currently held. During the process, the trading volume limit that can be initiated can be confirmed through "holding query", and the current recorded trading information can be checked through "backtracking bill" to confirm the signature and encryption status. Finally, the results of PoW and cipher algorithm are verified by "block check".

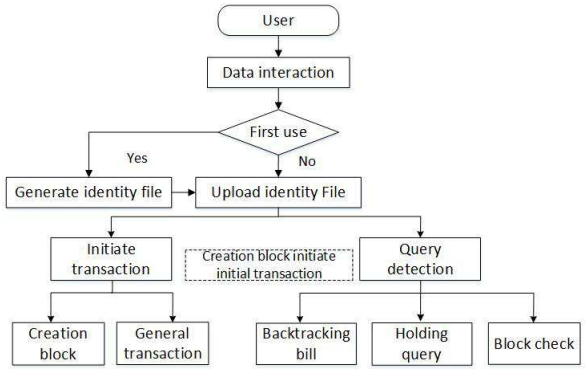


Figure 8: The flow of using

6.3 Access Control and Boundary Handling

This section describes in detail the access control and boundary handling policies adopted by function modules in the verification system. Data interaction module. When creating an identity file, the user's submitted identity cannot be empty. When a user submits an identity file, the content should correspond to what was encapsulated when the identity file was generated. If the identity file is missing or inconsistent, it will be judged illegal and users cannot access the transaction initiation module and query detection module.

Transaction initiation module. "Sender" is fixed and cannot be modified, especially the user ID read from the identity file. "Receiver" is the user id corresponding to the added node, and the server traverses the node information to detect it. If the "recipient" does not exist, the transaction is filtered and an error report is sent. Use regular expressions to determine the "amount" and "lost" input data types. Because the quality of industrial hemp corresponds, transactions are allowed to initiate if and only if "quantity" and "loss" inputs are real Numbers. In particular, the trigger condition of the "trade" and "Start" buttons is that all five input fields are not empty, that is, the transaction information filled in by the user must be complete.

Query detection module. In "block check", the detection contents related to blocks include workload proof and hash of the previous block. If the calculated results are inconsistent with the recorded content, the blockchain is not trusted. Transaction related detection content includes signature, encryption results. First, according to "sender" and "receiver", the signature and encryption principal are detected as registration nodes. If the subject does not correspond to the node information, the transaction will not be trusted. After that, the signature is verified.

7 Experimental Verification

7.1 PoW Test Analysis

Target hash: The number of 0's at the end is l.

Hit counts: PoW algorithm hits the target hash for the time of n.

Iterations: m represents the number of iterations of the PoW algorithm from the time n-1 to the time of n hit.

Intervals: k is the calculated increment used by the PoW algorithm.

This paper designs PoW algorithm based on SM3. Establish the blockchain private chain, cancel the incentive mechanism, require the new block mandatory generation. When a new transaction is initiated, if the record of the current number in transactions is p, it begins to calculate the work. In order to ensure that the number of transactions recorded in each block is similar and the block size is controlled at the same time, the PoW algorithm takes a stable hit as the main target and does not require excessive complexity.

This section measures and analyzes the hit of PoW algorithm under different conditions of k and l. In the experiment, $k \in \{1, 2, 3\}$, whose value is the prime factor of most natural numbers, can fully reflect the general influence of intervals on hit times. When the target is L, m_k is the number of iterations of the PoW algorithm. Record the measured value of m_k while l=1,$k \in \{1, 2, 3\}$, and part of the results are shown in Table 5.

As can be seen from Table 5, when l=1, the variation trend of m_k is similar for different k. That is, with the increase of hit times, the number of iterations increases and finally tends to be stable. Take l=2 and l=3 for multiple experiments, record data, and draw the curve of m

Table 5: The number of iterations under the hash of different targetse

N	m_1/times	m_2/times	m_3/times
1	11	11	88
2	33	33	106
3	56	87	106
4	59	107	106
5	73	127	106
6	75	129	106
7	78	165	106
8	79	187	106
9	82	211	106
10	123	243	106
11	125	273	106
12	137	297	106
13	144	325	106
14	157	333	106
15	166	347	106
16	190	381	106
17	196	441	106
18	204	491	106
19	227	493	106
20	240	597	106
21	259	711	106
22	272	743	106
23	311	743	106
24	313	743	106
25	333	743	106
26	347	743	106
27	381	743	106
28	406	743	106
29	432	743	106
30	435	743	106
31	435	743	106

Table 6: The number of hits required to achieve a constant time

N	l=1	l=2	l=3
k=1	30	71	6310
k=2	22	96	1759
k=3	2	112	379

changing with n, as shown in Figure 9. Where (a) is the changing trend of m with n when L =1; (b) is the changing trend of m with n when l=2; (c) is the changing trend of m with n when l=3.

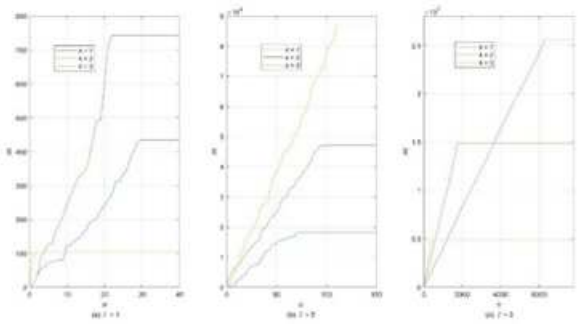


Figure 9: The change of iteration number under different target hashing

It can be seen from Figure 8 that the curve trend is increasing and finally reaches the level. In other words, for different k and l, the PoW algorithm hits a certain number of times, the target hash is uniformly distributed, and the number of iterations tends to be stable. This means that the blockchain using the PoW algorithm in this paper will have the following characteristics: as the chain length increases, the interval time required for the workload proof from the trigger of the calculation condition to the final generation of the new block gradually increases and finally reaches a constant. The different k and l only affect the hit number N needed to reach constant time, as shown in Table 6.

It can be seen from Table 6 that, when k is constant, N increases with the increase of l, and vice versa. This means that a blockchain using this PoW algorithm would have the following properties: the number of blocks that need to be generated before reaching a constant time can be controlled, and only by changing the target hash. The total number M_k of iterations needed to hit N times under different k values when l=3 is calculated by the following formula :

$$M_1 = \sum_{n=1}^{6310} m_1 = 81173004675$$

$$M_2 = \sum_{n=1}^{1759} m_2 = 12908482065$$

The minimum value is M_3, so the order of iteration required for N hits when l=4 is estimated:

$$M_3 = \sum_{n=1}^{379} m_3 = 909839005$$

It can be seen that when l=4, the number of iterations required to hit N times will be much larger than 9×10^8. Considering the time complexity of the SM3 algorithm, the iterative operation of the above order will consume huge computing resources, exceeding the computing capacity of personal computers. The block generation time

cannot be stable in a short time, which does not meet the practical application requirements.

To sum up, this paper designs a PoW algorithm based on SM3, and the blockchain using this algorithm will have the following two characteristics. First, when the blockchain reaches a certain length, the time taken to generate the block will tend to be stable. Second, the stable block generation time can and only can be achieved through target hash control. Based on the experiment in this section, the parameter values k=3,l=1 in the verification system were determined, so that the block generation time in the verification system could reach stability as soon as possible, and the calculation amount required for each hit was moderate, which could meet the requirements.

7.2 SM3 Test Analysis

Sample: Input data for any finite length string, password hash algorithm.

Sample length: l is the byte occupied by the sample.

Sample size: n is the total number of samples in the experimental group.

This paper designs the python software optimization implementation of the SM3 hash algorithm. In scheme A, the sample is processed as hexadecimal byte string directly, the cyclic left displacement is realized by the number theory method, and the message filling is optimized by mathematical reasoning. Scheme B is derived from gmssl-python and is used for comparison without computational optimization by mathematical reasoning. Scheme C is used for comparison, and the sample is transcoded to 0,1 string, and the operation is completed with data structures such as lists.

Due to the influence of hardware condition, external environment, and other factors, the code running time fluctuates in a small range. Therefore, the experimental group was designed, and the hash function was called for n times in the group to reduce the error. The contents of the 6 experimental groups are as follows:

Group one: n = 10, l = 100B, the samples are identical and consist of numbers, letters, and Chinese characters.

Group two: n = 15, l = 600B, the samples differ from one another, including five strings of numbers, five strings of letters, and five strings of Chinese characters.

Group three: n = 30, $l \in \{60B, 120B, 180B, \cdots, 600B\}$, the samples are different from each other, including 10 strings of numbers, 10 strings of letters, and 10 strings of Chinese characters. Different samples of the same kind appear alternately, and the length of the samples of the same kind increases by 60B.

Table 7: Time records for each scheme

	t_A/ms	t_B/ms	t_C/ms
Group One	9.973	10.935	69.814
Group Two	80.784	87.767	628.289
Group Three	108.357	123.633	715.047
Group Four	277.258	328.122	2045.533
Group Five	339.090	386.934	2581.123
Group Six	1042.207	1238.683	7613.630

Group four: n = 90, $l \in \{60B, 120B, 180B, \cdots, 600B\}$, the samples are different from each other, including 30 strings of Numbers, 30 strings of letters, and 30 strings of Chinese characters. Different samples of the same kind appear alternately, and the length of the samples of the same kind increases by 60B.

Group five: n = 60, $l \in \{60B, 120B, 180B, \cdots, 1200B\}$, it includes 20 different samples, each of which occurs 3 times in random order. The length of the sample increases by 60B and is composed of numbers, letters, and Chinese characters.

Group six: n = 1, l = 86079B, letters, Chinese characters, symbols, and English appear randomly in the sample.

The experimental encoding environment is UTF-8. t_A,t_B,t_C is the time using in each scheme and they were recorded respectively, as shown in Table 7. Scheme A takes the shortest time to complete 6 groups of experiments.

Compared with Scheme B, Scheme A has a stronger compressive resistance and is more advantageous in dealing with long samples. D_i is denoted as the difference between the time taken by Scheme B and Scheme A to process per byte in the experiment of group i. In group 3, n = 30, D_3=1.543 us was calculated. In group 4, n = 90, other conditions being equal, D_4=1.713 us was calculated, and it was found that scheme A took less time per byte when processing multiple samples continuously.

$$D_3 = \frac{t_{B_3} - t_{A_3}}{\sum_{i=0}^{n} l_i} = \frac{123.633 - 108.357}{9900} = 1.543\,us$$

$$D_4 = \frac{t_{B_4} - t_{A_4}}{\sum_{i=0}^{n} l_i} = \frac{386.934 - 339.090}{34140} = 1.401\,us$$

In group 6, l = 86079B was the maximum sample length used, D_6=2.283 us, and the difference of processing time per byte was also the maximum.

$$D_6 = \frac{t_{B_6} - t_{A_6}}{\sum_{i=0}^{n} l_i} = \frac{1238.683 - 1042.207}{86079} = 2.283\,us$$

Compared with plan C, Scheme A significantly reduces the time complexity. $\overline{E}_{AC}$ is the average ratio of time taken to complete each experiment in Scheme A and C. Scheme A deals with Bytes while Scheme C deals with

Table 8: The time taken of the derived key for each scheme

	$\bar{t}_A/ms$	$\bar{t}_B/ms$
d_1	158.50	227.69
d_2	190.36	262.39
d_3	204.84	294.60
d_4	177.11	340.47
d_5	190.48	260.33
d_6	176.18	294.58

bits. $\overline{E}_{AC} \equiv 0.138$ accords with the theoretical values and reflects the algorithm optimization.

$$\begin{aligned} \overline{E_{AC}} &= \frac{1}{6}\sum_{i=0}^{6}\frac{t_{A_i}}{t_{C_i}} \\ &= \frac{1}{6}(0.143+0.128+0.152+0.131+0.137) \\ &\approx 0.138. \end{aligned}$$

In conclusion, the SM3 hash algorithm python software implementation scheme designed in this paper reduces the computation time and achieves a better optimization effect. At the same time, from the perspective of the system, the SM3 software implementation scheme designed in this paper has a certain compressive ability, which can meet the practical application requirements of blockchain.

7.3 SM9 Test Analysis

This paper designs a python software optimization implementation of the SM9 identification cipher algorithm. This section will design experiments for key derivation, signature verification, encryption, and decryption to verify the optimization effect. Scheme A, based on bytes structure in python, is used to process data in byte bits, and integer type is used to complete bitwise ecotone or other Boolean operations. At the same time, the key export and hash function are optimized by mathematical reasoning. Scheme B is used for comparison. The operation process involves character data, and the calculation is not optimized by mathematical reasoning.

Take the signature private key as an example to measure the key derivation time of schemes A and B. In the experiment, the user id was randomly generated, with a total of 6 strings of characters within 20 in length, denoted as D= d_1,d_2,d_3,d_4,d_5,d_6 . Due to the influence of hardware condition, external environment, and other factors, the code running time fluctuates in a small range. Therefore, for $\forall d_i \in D$, adopt measures of average 6 times to reduce error. The time taken for each scheme to generate the signature private key was measured, and the average ta tb was calculated. The results are shown in Table 8: The signature and check algorithms were respectively executed to measure the operation times per minute of scheme A and B. The experimental encoding environment is UTF-8. The pending information is the same. For the signature algorithm, the 256 bit length SM3 hash calculation result is used, which directly participates in the signature operation to eliminate the interference of the hash algorithm to the experiment. For the signature check algorithm, the signature used is the above information. Execute continuously and record the completion times of each scheme per minute c_A, c_B, as shown in Table 9. The encryption

Table 9: Comparison of operation times of signature check

	c_A/tpm	c_B/tpm
signature	336	304
signature check	62	59

and decryption algorithms are respectively implemented to measure the encryption and decryption rates of scheme A and B. Randomly generate 10 strings of characters in bytes $N*10^3$,$N \in$ {1,2,3...9,10}. For $\forall N$, we use the program group key, and alternate with six groups of encryption and decryption operations, each time of encryption operation is recorded as e_i,and each time of decryption operation is recorded as d_i,$i \in [1,6]$. Computing the encryption number of bytes per second, and the encryption speed $v = 6*N*10^3/\sum_{i=1}^{6} e_i$. Computing the decryption number of bytes per second, and the decryption rate $w = 6*N*10^3/\sum_{i=1}^{6} e_i$, the results as shown in Table 10:

It can be seen from Figure 10 that the encryption and decryption rates of scheme A are higher than those of Scheme B. Both v_A,w_A remain stable and do not change significantly with the increase of plaintext length. Furthermore, calculating the average $\overline{v}_A = 614.26$,$\overline{w}_A = 370.22$ standard deviation $\sqrt{D(v_A)}$ =31.90,$\sqrt{D(p_A)}$ =23.23. It is difficult for an attacker to infer the length of a key by changing the rate of encryption and decryption.

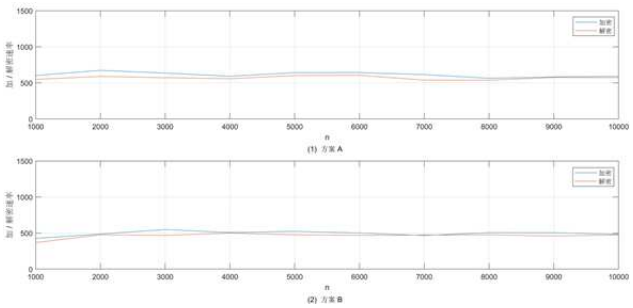


Figure 10: The encryption and decryption rates vary with the length of plaintext

Diffusion requires the relationship between Ming and ciphertext to be as complex as possible, and any small change in the plaintext will make the ciphertext greatly different [9]. Experiments are designed to verify that the security of the encryption algorithm itself is not damaged by the optimization of scheme A. Randomly generate a string of characters of length 100.3 groups of experiments

Table 10: Encryption and decryption rate of each scheme

N	$v_A/(B*s^{-1})$	$w_A/(B*s^{-1})$	$v_B/(B*s^{-1})$	$w_A/(B*s^{-1})$
1	602.05	548.72	426.08	368.30
2	673.63	590.69	486.54	475.37
3	636.12	571.77	547.85	467.43
4	592.52	558.54	507.40	500.41
5	641.66	601.28	523.50	476.30
6	642.47	605.85	500.34	467.72
7	615.41	538.48	466.77	473.19
8	564.05	537.23	505.58	476.03
9	583.13	576.28	504.51	458.77
10	591.59	573.38	482.50	475.33

were conducted to randomly generate characters and replace k in the plaintext,$k \in\{1,2,3\}$. For $\forall k$, six experiments, each plaintext character have the same number change, but the content and position of random. The mean value and standard deviation of character changes are calculated. The discrete distribution is shown in Table 11: In the experiment, the change of plaintext and

Table 11: The ciphertext character changes of scheme A

k	mean ± SD
1	93.50 ± 1.50
2	95.67 ± 1.49
3	94.50 ± 3.10

ciphertext will have great changes. The number of ciphertext character changes is above 90, accounting for more than 90% It can be seen that the plaintext diffusion is good, and the optimization of scheme A does not destroy the security of the algorithm.

To sum up, this section designs experiments to verify the software optimization effect of python for the SM9 identification cipher algorithm. In the key derivation, it takes less time to generate the signature private key with different user identities. In signature verification, experiments were conducted respectively, and the signature and verification algorithms were successively executed, with more operations per minute. In encryption and decryption, the plaintext length is different, multiple encryption and decryption operation, the rate is faster. At the same time, plaintext diffusion is good, security is not damaged.

Conclusion

At present, the supervision of industrial hemp has the problems of low efficiency and low safety. This paper analyzes the actual needs of industrial hemp monitoring and proposes a solution based on the technical background of blockchain and SM series algorithms. First, We use the private chain based on PoW, remove the incentive mechanism. Then we design a block and trading structure according to industrial hemp monitoring requirements. In the block record content, "hash of previous block" is generated by the SM3 hash algorithm to protect the integrity of the block. In the transaction record content, "signature" and "type" are generated by the SM9 identification algorithm respectively, which protect the authenticity and confidentiality of the transaction. In the design, the creation block and the initial transaction have a special nature and are initiated only by the producer. In the initial transaction, the "quantity" corresponds to the total quality of the batch of industrial hemp. Taking the initiation node of the Genesis block as the super source node and the transaction as the arc, a directed graph is constructed to monitor three types of anomalies which are the illegal recipient, insufficient holding, and excessive loss.

In the software implementation, SM3 and SM9 are optimized with the help of mathematical reasoning and Python3 bytes structure, which reduces the computational overhead.

The verification system was established, and the data interaction module, transaction initiation module, and query detection module are corresponding to various functions of the blockchain. It is supplemented by access control and boundary treatment strategies, which serve as an experimental verification platform with complete functions,a beautiful interface, and friendly interaction. In the experiment, in addition to testing various functions, PoW computing overhead and SM3 and SM9 optimization were also tested. PoW calculation overhead is stable and the number of transactions recorded in each block is similar. The optimization effect of SM3 and SM9 is obvious, the time and space overhead of hash, key generation, signature, check, encryption, decryption, and other calculation are reduced, and the security is not broken.

References

[1] T. M. Attard, C. Bainier, M. Reinaud, *et al.*, "Utilisation of supercritical fluids for the effective extraction of waxes and cannabidiol (CBD) from hemp wastes," *Industrial Crops and Products*, vol. 112, pp. 38-46, 2018.

[2] G. Baochang, S. Yufeng, Z. Xu, *et al.*, "Study on the content of cannabidiol in hemp leaves," *Heilongjaing Science*, vol. 9, no. 01, pp. 61-63, 2018.
[3] Z. Fake, *Method and Process for Producing High-Purity Tetrahy Drocannabinol by Chromatography*, CN201910619293.4, Sept. 13, 2019.
[4] K. Jiaqian, Z. Mingsen, M. Xiaokang, G. Jinhu, F. Xuping, K. Hongmei, "Adaptability analysis of different industrial hemp varieties in Shanxi," *Shanxi Agricultural Sciences*, vol. 47, no. 10, pp. 1803-1805, 2019.
[5] W. Jufeng, Y. Xiaoquan, "Optimization of enzyme-assistedsolvent extraction technologyofcannabidiol from hemp leaf," *China Brewing*, vol. 35, no. 04, pp. 79-82, 2016.
[6] H. Miyano, "Addend dependency of differential, linear probability of addition," *IEIcE Trans on Fundamentals of Electronics, Communications and Computer Sciences*, vol. E81-A, no. 1, pp. 106-109, 1998.
[7] C. Ping, T. Siying, "Research on intelligent financial reporting based on blockchain technology," *Friends of Accounting*, no. 05, pp. 156-160, 2020.
[8] W. Rihong, Z. Lifeng, Z. Hang, *et al.*, "A byzantine fault tolerance raft algorithm combines with BLS signature," *Journal of Applied Sciences*, vol. 38, no. 01, pp. 93-104, 2020.
[9] C. E. Shannon, "Communication theory of secrecy systems," *Bell System Technical Journal*, vol. 28-4, pp. 656–715, 1949.
[10] Standardization Administration, *Information Security Techniques — SM3 Cryptographic Hash Algorithm*, Standardization Administration, GB/T 32905-2016, 2016.
[11] S. Taoyi, Z. Yunlei, "Comparison of blockchain consensus algorithm," *Computer Applications and Software*, vol. 35, no. 08, pp. 1-8, 2018.
[12] Md A. Uddin, A. Stranieri, I. Gondal, V. Balasubramanian, "Blockchain leveraged decentralized IoT eHealth framework," *Internet of Things*, vol. 9, 2020. DOI:10.1016/j.iot.2020.100159.
[13] H. Wang, H. Qin, M. Zhao, X. Wei, H. Shen, W. Susilo, "Blockchain-based fair payment smart contract for public cloud storage auditing," *Information Sciences*, vol. 519, pp. 348-362, 2020.
[14] G. Xianjin, X. Juncheng, *Method for Extracting Tetrahydrocannabinol from Medical Marijuana*, CN108654134A, Oct. 16, 2018.
[15] W. Xiaorui, Z. Xuechao, "Research and application of blockchain security technology," *Network Security Technology & Application*, no. 02, pp. 29-30, 2020.
[16] L. Yanxu, D. Peng, C. Xiaoguang, *et al.*, "An approach for the analysis of pharmacodynamic interactions and the simulation of combined response," *Chinese Pharmacological Bulletin*, no. 08, pp. 1112-1114, 2007.
[17] Y. Yatao, C. Juliang, Z. Xiaowei, *et al.*, "Privacy preserving scheme in block chain with provably secure based on SM9 algorithm," *Journal of Software*, vol. 30, no. 06, pp. 1692-1704, 2019.
[18] C. Yunchun, L. Haonan, "Study on the construction of reverse supply chain for expired drugs under block chain perspective," *China Pharmacy*, vol. 30, no. 24, pp. 3342-3349, 2019.
[19] G. Zhe, Z. Zhijun, L. Xiaojun, *et al.*, "Hot reflux extraction of cannabidiol from hemp leaves," *China Oils and Fats*, vol. 44, no. 03, pp. 107-111, 2019.
[20] B. Zhenshan, W. Kaixuan, Z. Wenbo, "A practical byzantine fault tolerance consensus algorithm based on tree topological network," *Journal of Applied Sciences*, vol. 38, no. 01, pp. 34-50, 2020.

Biography

The First Author biography. Zijian Ma received the B.S. degree in 2007 and Master degree in 2009 in Industrial engineering from Tsinghua University. He is pursuiting his PH.D degree of Computer and Information Technology in Beijing Jiaotong University now. His research interest includes artificial intelligence and data mining.

The Second Author biography. Zhiqiang Wang, born in China in 1985. He received the Ph.D. degree in information security from Xidian University.He is currently an Assistant Professor with the Department of Computer Science and Technology. His research interests include system security and network security.

The Third Author biography. Hang Wu, born in 1995, is currently a graduate student at Xidian University, majoring in cryptography. Her research direction is hardware Trojan detection technology based on convolutional neural network.

Other Author biography.
Xingyu Guo,he is a undergraduate student in Beijing Electronic Science and Technology Institute.His research direction is network security. Xizhen Wang, born in 1995, is currently a graduate student at Xidian University, majoring in cryptography. His research direction is the research of lattice-based signature schemes.

Security Situation Prediction Method for Industrial Control Network Based on Adaptive Grey Verhulst Model and GRU Network

Rui-Hong Dong, Chuang Shu, Qiu-Yu Zhang, and Ya-Yu Mo
(*Corresponding author: Qiu-Yu Zhang*)

School of Computer and Communication, Lanzhou University of Technology
No. 287, Lan-Gong-Ping Road, Lanzhou 730050, China
Email: zhangqylz@163.com

(Received Dec. 22, 2020; Revised and Accepted May 6, 2021; First Online Nov. 9, 2021)

Abstract

Aiming at the problems of existing industrial control network security situation prediction methods that the prediction accuracy is insufficient, the model is difficult to construct and requires a large amount of training data. Therefore, industrial control network security situation prediction method based on adaptive grey Verhulst model and gated recurrent unit (GRU) network was proposed. Firstly, the background value generation of the traditional grey Verhulst model is improved to adapt to the nonlinear character of historical and current network security situation sequence. Secondly, the historical and current network security situation value sequence is input into the improved grey Verhulst model to obtain the preliminary prediction value and residual sequence of the network security situation. Then, the preliminary prediction value sequence is taken as the input and the residual sequence as the output to train the GRU network. Finally, the trained GRU network is used to predict the residual error and output the predicted value of the network security situation after the residual error correction. Experimental results show that the proposed method can reach the convergence state faster than the existing prediction methods. Moreover, the loss value is smaller, which has a higher accuracy of network security situation prediction.

Keywords: Adaptive Grey Verhulst Model; GRU Network; Industrial Control Network; Network Security Situation Prediction; Residual Sequence

1 Introduction

With the proposed strategies of "Industry 4.0", "Industrial Internet" and "Made in China 2025", the integration of all aspects of industrialization and informatization continues to deepen and develop. In this process, the traditional industrial system is gradually developing from the physical isolation control system in the past to the multi-dimensional control of the Internet. The hidden network security risks in the Internet have gradually transitioned to the industrial control system, causing the industrial control network to face huge security threats [3]. In recent years, new types of network attack technologies have emerged one after another, the problem of network vulnerability has become increasingly prominent, and network information security is facing severe challenges [2]. Therefore, network security situation prediction plays an important role in network defense, network security early warning and network resource allocation, and has become a research hotspot in the field of industrial control network security [12].

Prevention is more effective than detection and recovery. With the increasing number of industrial network security threats, network administrators are eager to have a clear understanding of their network security situation before taking defensive measures [11]. Network security situation prediction is based on the historical and current network security situation assessment to further predict the development trend of possible network security in the future, which can help the network administrators have more time and preparation to cope with the possible arrival of threats, reasonably allocate network resources, and take accurate preventive measures against the target network [27]. Network security situation prediction mainly focuses on the fitting of nonlinear situation sequence. There are many existing network security situation prediction methods, such as neural network, support vector machine (SVM), Markov model, grey theory, belief rule-base and so on. In [23], an improved particle swarm optimization (PSO) algorithm is adopted to optimize the radial basis function (RBF) neural network for network security situation prediction and to improve the prediction accuracy and convergence speed of the prediction algorithm.

Zhang *et al.* [24] proposed a BP neural network secu-

rity situation prediction method based on hybrid rice algorithm optimization. Through the advantages of global search and rapid convergence of hybrid rice optimization algorithm, the future security situation of the network was predicted, which effectively improved the accuracy of situation prediction. Han *et al.* [5] put forward a network security situation prediction method based on intuitionistic fuzzy sets of nonlinear regression neural network (IFS-NARX), which solves the problem of existing prediction methods are difficult to deal with network security situation prediction uncertainty under the complex network environment and using empirical data effectively improve the efficiency and accuracy of network security situation prediction. Shang *et al.* [27] brought up a network security situation prediction method based on LSTM-XGBoost model. The improved LSTM neural network model is used to predict the network security data, and then the XGBoost model is utilized to evaluate the situation of the predicted data. As traditional network security situation prediction methods rely on the accuracy of historical situation values, He *et al.* [6] brought forward a GRU coding prediction method based on attention mechanism. This method uses GRU neural network to mine the time correlation between network security situation data. Attention mechanism is introduced to calculate the assigned weight of security index and encode it into network security situation value. As the existing haze concentration prediction model cannot accurately capture the law between haze concentration and influencing realistic factors, it is difficult to accurately predict the nonlinear haze data.

Wang *et al.* [21] proposed a two-layer model prediction algorithm based on Long Short Term Memory Neural Network and Gated Recurrent Unit (LSTM&GRU). Chen *et al.* [2] proposed a network security situational awareness algorithm based on regression predictive SVM (RF-SVM). This algorithm adopts the regressive thought, fully refers to the historical network attack data in the network perception process, and predicts the potential threat in the future network data stream. Chen *et al.* [20] introduced a reliability prediction model of SVM based on genetic algorithm optimization, aiming at the current situation of inaccurate single prediction model of network security situation prediction. Hu *et al.* [14] proposed a network security situation prediction model based on MapReduce and SVM to solve the problem of long training time of SVM. This model make use of cuckoo search algorithm (CS) for parameter optimization to determine the optimal parameters of SVM, and chooses MapReduce for distributed training on SVM to improve the training speed. Liang *et al.* [26] presented a multi-scale entropy weighted hidden Markov prediction model. The model adopts the multi-scale entropy method to select appropriate scale factors as training data, and employ the correlation coefficient weighting method to predict the future security situation.

Yang *et al.* [25] come up with a new method of network security situation prediction on the basis of self-correcting coefficient smoothing method, this method establishes multiple coefficient of smoothing model to get the initial forecast, through the initial forecast deviation based on the real consequences and time-varying weighted Markov chain, through the model to forecast the deviation and correct the initial forecast, the resulting network security situation prediction results. In view of the character of intrusion attacks and the deficiency of traditional grey Verhust model, Leau *et al.* [11] proposed an adaptive grey Verhust model with adjustable generating sequence to improve the prediction accuracy. Aiming at the existing network security prediction methods in cloud environment have some limitations in accuracy and real-time, a network security prediction method based on grey neural network is proposed in [16]. Hong *et al.* [7] brought up a network security situation prediction method based on grey relational analysis and SVM algorithm. The grey relational analysis theory is used to weight network evaluation indexes, and the prediction process is simulated based on SVM algorithm. A network security situation prediction model based on multi-swarm chaotic particle optimization and optimized grey neural network is proposed in [17]. By establishing the nonlinear mapping relationship between network evaluation index and network security situation, the network situation can be predicted.

At present, the research on network security situation prediction in the domestic and overseas is mainly based on data-driven, and a prediction model that can reflect the hidden behavior characteristic values is directly established by using observable data and corresponding methods [22]. With the development of new technologies such as deep learning and machine learning, network security prediction technology is developing towards intelligence and integration [9]. The latest research direction is to combine machine learning method with Grey theory, Markov model and other forecasting methods, and to predict the complex situation change of industrial control network through multi-layer nonlinear abstract fitting, which needs to be tested and improved in practice, both in feasibility and optimality.

By analyzing the above research, existing research have provided a series of feasible solutions for network security situation prediction, but there are still some problems, such as insufficient prediction accuracy, difficulty in model construction, and the need for a large amount of training data. To solve the above problems, this paper proposes a security situation prediction method for industrial control network based on adaptive grey Verhulst model and GRU network. The main innovations of this paper are as follows:

1) Analyze the error source of the background value of the traditional grey Verhulst model through the combination of number and shape, and reconstruct the generation of the background value by the combination method of the complex trapezoidal rule and the complex Simpson's rule, so as to adapt to the nonlinear characteristics of the historical and current network security situation sequence;

2) The improved grey Verhulst model is used to make preliminary prediction of network security situation sequence, and the trained GRU network is used to correct the residual error of the preliminary predicted value, thus solving the problem that the prediction model is difficult to build;

3) The improved grey Verhulst model and GRU network are combined to predict the network security situation sequence, and the poor information of the grey Verhulst model is used to replace the large sample required by the neural network, and the non-linear processing capacity of GRU network is used to compensate for the poor nonlinear fitting of the grey Verhulst model, thus improving the accuracy of network security situation prediction.

The remaining part of this paper is organized as follows. Section 2 introduces the relevant theories of network security situation prediction. Section 3 describes in detail the improvement of grey Verhulst model and the process of predicting network security situation value by combining GRU network. In Section 4, the security situation prediction method is experimentally verified and compared with the existing methods. Section 5 concludes the presented work and raises several issues of future work.

2 Relevant Theories

2.1 Network Security Situation Factor

In order to achieve the accurate prediction of network security situation, based on the vulnerability information existing in the network, aiming at the complexity and heterogeneity of network operation data, as well as the relevance of various impact indicators, and analyzes the impact of attack on situation assessment and proposes a situation index assessment factor. Including: attack quantity, attack frequency, attack threat, vulnerability quantity, vulnerability threat, vulnerability probability of exploitation, and other six factors, and give the relevant definition.

Definition 1. *Attack Quantity Factor: Defined as the number of attack types detected within a certain period of time, denoted as N.*

Definition 2. *Attack Frequency Factor: Defined as the attack frequency of each type of attack in a certain period of time, denoted as C_i (i represents different attack types).*

Definition 3. *Attack Threat Factor: Different attack types have different degrees of impact on network security operations, denoted X_i (i represents different attack types).*

Definition 4. *Vulnerability Quantity Factor: Defined as the number of vulnerability types scanned on a network node, denoted as M_{id_v}.*

Definition 5. *Vulnerability Threat Factor: Defined as the impact of each vulnerability on the normal operation of the network, denoted as $impact_v$.*

Definition 6. *Probability Factor: Defined as the probability of each vulnerability being successfully exploited, denoted as p_{suc}.*

2.2 Grey Verhulst Prediction Model

The Verhulst model is a single-sequence first-order nonlinear dynamic model [19]. It is mainly used to describe the process with saturation state, namely S-shaped process.

Set $x^{(0)}$ as the network security situation value sequence,

$$x^{(0)} = (x_1^{(0)}, x_2^{(0)}, \cdots, x_n^{(0)}).$$

$x^{(1)}$ is the one-time accumulation of $x^{(0)}$ to generate sequence (1-AGO),

$$x^{(1)} = (x_1^{(1)}, x_2^{(1)}, \cdots, x_n^{(1)}),$$

where $x_k^{(1)} = \sum_{i=1}^{n} x_i^{(0)}$ $(k = 1, 2, \cdots, n)$.

$z^{(1)}$ is the sequence generated by the immediate mean value of $x^{(1)}$,

$$z^{(1)} = (z_1^{(1)}, z_2^{(1)}, \cdots, z_n^{(1)}),$$

where $z_k^{(1)} = 0.5 \cdot (x_k^{(1)} + x_{k-1}^{(1)})$.

Equation (1) is the grey Verhulst model.

$$x^{(0)} + a \cdot z^{(1)} = b \cdot (z^{(1)})^2, \tag{1}$$

where, a is the development coefficient, and its size reflects the growth rate of series $x^{(0)}$; b is the role of grey input in the grey Verhulst model.

Equation (2) is the whitening equation of grey Verhulst model, where t is time.

$$\frac{dx^{(1)}}{dt} + a \cdot x^{(1)} = b \cdot (x^{(1)})^2. \tag{2}$$

The grey Verhulst model is assumed as described above, if $\hat{\alpha} = (a, b)^T$ is the parameter column, and

$$B = \begin{bmatrix} -z_2^{(1)} & (z_2^{(1)})^2 \\ -z_3^{(1)} & (z_3^{(1)})^2 \\ \vdots & \vdots \\ -z_n^{(1)} & (z_n^{(1)})^2 \end{bmatrix}, \quad Y = \begin{bmatrix} -x_2^{(0)} \\ -x_3^{(0)} \\ \vdots \\ -x_n^{(0)} \end{bmatrix}$$

Then the least square estimation of parameter column satisfies the condition of Equation (3):

$$\hat{\alpha} = (B^T B)^{-1} B^T Y. \tag{3}$$

The solution (time response function) of the whitening equation is as follows.

$$x^{(1)}(t) = \frac{a x_0^{(1)}}{b x_0^{(1)} + (a - b x_0^{(1)}) \cdot e^{at}}$$

Equation (4) is the time response sequence of grey Verhulst model.

$$\hat{x}_{k+1}^{(1)} = \frac{ax_0^{(1)}}{bx_0^{(1)} + (a - bx_0^{(1)}) \cdot e^{ak}} \tag{4}$$

where, if $x_0^{(1)}$ is equal to $x_1^{(0)}$, Equation (4) is converted to Equation (5):

$$\hat{x}_{k+1}^{(1)} = \frac{ax_1^{(0)}}{bx_1^{(0)} + (a - bx_1^{(0)}) \cdot e^{ak}} \tag{5}$$

The reductive formula is shown in Equation (6):

$$\hat{x}_{k+1}^{(0)} = \hat{x}_{k+1}^{(1)} - \hat{x}_k^{(1)} \ (k = 2, 3, \cdots, n). \tag{6}$$

where $\hat{x}_{k+1}^{(0)}$ is the predictive value sequence of network security situation value sequence $x^{(0)}$.

2.3 GRU Network Model

The gated recurrent unit (GRU) [4] was proposed by K. Cho *et al.* in 2014. It can be regarded as a simplified version of long short-term memory (LSTM) [27]. Both use the gating mechanism to retain the useful information of the previous sequence, ensuring that it will not be lost during long-term propagation, which is suitable for processing and predicting important events with relatively long intervals and delays in the time series. GRU maintains the LSTM effect while making the structure simpler. The GRU network model is shown in Figure 1.

As can be seen from Figure 1, the GRU network model is divided into input layer, coding layer and output layer from bottom to top.

Input Layer: The network security situation sequence is processed into a data structure suitable for GRU network training.

Coding Layer: GRU can capture the interdependence among time series by introducing gating mechanism and controlling the flow of information adaptively. GRU has two gates, namely reset gate and update gate. r_t represents the reset gate and z_t represents the update gate. W_z represents the weight matrix of the update gate, and W_r represents the weight matrix of the reset gate. The reset door determines whether the previous state is forgotten. When r_t approaches 0, the state information h_{t-1} at the previous moment will be forgotten, and the hidden state $\tilde{h}_t$ will be reset to the current input information. The update gate determines whether the hidden state is to be updated to the new state h_t.

$$\begin{aligned} z_t &= \sigma(W_z \cdot [h_{t-1}, x_t]). \\ r_t &= \sigma(W_r \cdot [h_{t-1}, x_t]). \\ \tilde{h}_t &= \tanh(W \cdot [r_t * h_{t-1}, x_t]). \\ h_t &= (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t. \end{aligned}$$

Output Layer: Output the hidden state of the updated coding layer at different times.

3 The Proposed Prediction Method

3.1 Network Security Situation Prediction Method

The industrial control system communication networks usually contain a large number of hosts, network devices, and various detection systems that monitor the network from different perspectives and generate logs and alerts. Network security administrators can learn the security and stability of current network operations in time by analyzing these massive logs, alarms and other network data. Traditional network security situation prediction methods have insufficient prediction accuracy. In addition, the existing prediction models are difficult to build and require a large amount of training data, which directly affects the timeliness and effectiveness of prediction. As a result, network security administrators may make misjudgments about possible future network threats and take unreasonable network warning and defense measures. In response to these problems, this paper proposes a security situation prediction method for industrial control network based on the adaptive grey Verhulst model and GRU network. The poor information of the grey Verhulst model replaces the large samples required by the neural network, and the nonlinear processing capability of the GRU network make up for the shortcomings of the grey Verhulst model's poor nonlinear fitting, and establish a better performance grey Verhulst-GRU network prediction model. The processing flow of this prediction method is shown in Figure 2.

It can be seen from Figure 2 that the prediction method mainly includes the following prediction steps:

Step 1: Based on the text SimHash algorithm, the intrusion detection information is transformed into a nonlinear time series of network security situation values. Normalize the historical and current network security situation sequences to obtain prediction data samples $x_1^{(0)}$.

Step 2: According to the processed network security situation value sequence, an adaptive grey Verhulst prediction model is established, and the grey background value generation function is improved to improve the accuracy of preliminary prediction. Obtain the preliminary situation prediction value $\hat{x}_1^{(0)}(k+1)$, and calculate the residual $e_1^{(0)} = \left|x_1^{(0)} - \hat{x}_1^{(0)}\right|$ between the predicted value and the actual value.

Step 3: Determining the number of training samples and prediction samples. The predicted value is input to the input of the GRU neural network, the residual is the output result, and all training samples pass the mean square error of the GRU network as the goal to train the GRU network.

Step 4: The trained GRU network was used to predict the residual sequence, and the predicted residual

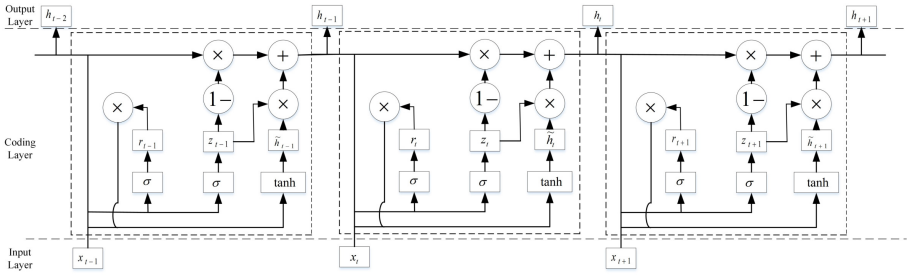


Figure 1: GRU network model

value $\hat{e}_1^{(0)}$ was obtained.

Step 5: The preliminary prediction results are revised to check the accuracy. If the requirements are not met, return to **Step 3**; Otherwise, output the final network security situation prediction value: $\hat{y} = \hat{x}_1^{(0)} + \hat{e}_1^{(0)}$.

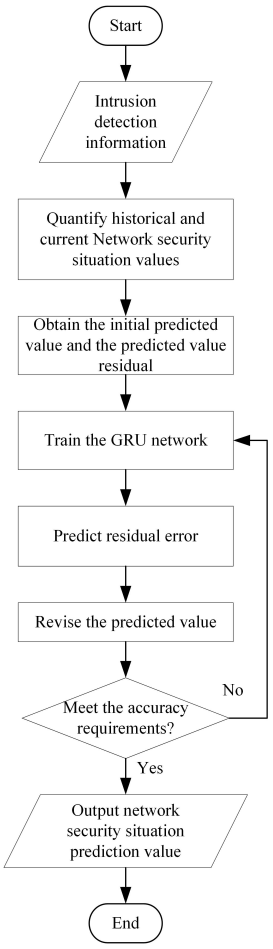


Figure 2: A network security situation prediction method based on adaptive grey Verhulst-GRU network

3.2 Improved Grey Verhulst Prediction Model

The industrial control network is a dynamic time-varying system, so the network security situation has a certain degree of random volatility, and its development presents a non-stationary random process with a certain changing trend [8]. Industrial control networks are facing a large number of malicious attacks. Although many of the attacks are large in scale, the severity of the attacks is very low. Therefore, the cumulative curve of the risk index grows slowly. But after this stage, the intruder began to realize the vulnerability and security level of the target. Aiming at the security level and vulnerability of the target, a stronger attack is adopted. This intrusion makes the cumulative curve of the risk index grow faster and the curve becomes steep. In the following process, the risk index reached its limit, after which the curve became smoother.

3.2.1 The Error Source of Background Value of Grey Verhulst Model

The security situation of industrial control network is represented by a saturated S-shaped process, so the Verhulst model in Section 2.2 can be used to predict it. The time response sequence Equation (5) of the grey Verhulst model is further simplified and can be reduced to Equation (7):

$$\hat{x}_{k+1}^{(1)} = \frac{1}{\frac{b}{a} + (\frac{1}{x_1^{(0)}} - \frac{b}{a})e^{ak}} \tag{7}$$

It can be seen from Equation (7) that the time response sequence function of the grey Verhulst model differential equation is in the form of Logistic function $x^{(1)}(t) = \frac{1}{p+qe^{m(t-1)}}$. Therefore, for the first-order cumulative generation sequence that is approximately the Logistic function, the traditional background value calculation formula $z_k^{(1)} = 0.5(x_k^{(1)} + x_{k-1}^{(1)})$ will bring certain errors. It essentially approximates the area of the trapezoid by replacing the area of the curve $x^{(1)}(t)$ on the interval $[k-1, k]$ with the t-axis. The shaded area in Figure 3 is the error portion.

Integrating both sides of the whitening Equation (2) of the grey Verhulst model on the interval $[k-1, k]$ at the same time, it can be obtained:

$$\int_{k-1}^{k} \frac{dx^{(1)}(t)}{dt} dt + a \int_{k-1}^{k} x^{(1)}(t) dt = b \int_{k-1}^{k} (x^{(1)}(t))^2 dt \tag{8}$$

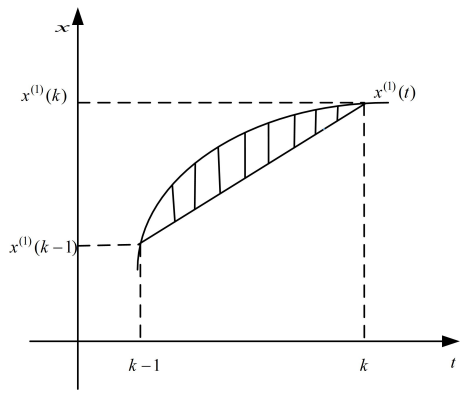


Figure 3: The error source of the background value of the traditional Grey Verhulst model

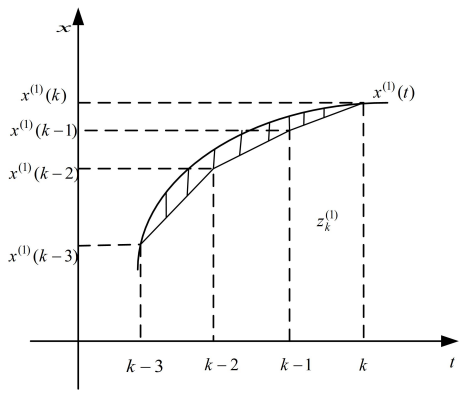


Figure 4: The area under the graph function

where,

$$\int_{k-1}^{k} \frac{dx^{(1)}(t)}{dt} dt = x^{(1)}(k) - x^{(1)}(k-1) = x^{(0)}(k).$$

Equation (8) is transformed into the following equation:

$$x^{(0)}(k) + a \int_{k-1}^{k} x^{(1)}(t)dt = b \int_{k-1}^{k} (x^{1}(t))^2 dt. \tag{9}$$

By comparing Equation (9) and Equation (1), it is found that when the first-order accumulation generation sequence is approximately Logistic function, $\int_{k-1}^{k} x^{(1)}(t)dt$ is used to replace the traditional background value $z_k^{(1)} = 0.5(x_k^{(1)} + x_{k-1}^{(1)})$, and replacing $(z_k^{(1)})^2$ with $\int_{k-1}^{k} (x^{(1)}(t))^2 dt$ to obtain the values of parameters a and b, which are more suitable for the whitening equation.

3.2.2 Reconstruction of Background Value of Grey Verhulst Model

It can be seen from Equation (7) that the accuracy of the grey Verhulst model depends on parameters a and b, and the values of a and b are directly affected by the construction form of $z_k^{(1)}$. In order to improve the performance of grey theory, especially in grey Verhulst model, it is necessary to eliminate the error term caused by background value. Therefore, it is an important task to find an appropriate background value $z_k^{(1)}$ for the model to improve the prediction accuracy.

Since the curve function $x^{(1)}(t)$ is unknown, the background value $z_k^{(1)}$ is calculated by the combined method of the complex trapezoid rule and the complex Simpson's rule in the adaptive grey Verhulst model.

From Figure 4, the area under the time interval from $k-1$ to k can be determined as follows:

$$z_k^{(1)} = \int_{k-1}^{k} x^{(1)}(t)dt = \int_{k-3}^{k} x^{(1)}(t)dt - \int_{k-3}^{k-1} x^{(1)}(t)dt.$$

Applying the complex trapezoidal rule and the complex Simpson's rule, the background value $z_k^{(1)}$ can be further simplified as Equation (10).

$$z_k^{(1)} = x^{(1)}(k-1) + \frac{1}{6}x^{(0)}(k-1) - \frac{1}{6}x^{(0)}(k-2) + \frac{1}{2}x^{(0)}(k) \tag{10}$$

By replacing the traditional background value generation method with the reconstructed background value calculation formula, the error caused by the background value can be eliminated. Substituting Equation (10) into the grey Verhulst model Equation (1) in Section 2.2 can improve the prediction accuracy of the grey Verhulst model for the security situation sequence value.

3.3 Correction of Predicted Value

The network security situation prediction results are initially generated based on the improved grey Verhulst model. According to experience, the prediction results deviate from the actual values of the known security situation. In order to further reduce this deviation, the optimized GRU network is used to perform residual correction on the preliminary prediction value.

3.3.1 Parameter Optimization

The GRU network training data in Section 2.3 involves the setting of multiple hyperparameters: number of neurons, time step size, and the batch size. The number of neurons determines the fitting degree of the neural network, and the time step and batch size directly affect the results of model training. In practical applications, different data corresponding to different hyperparameter settings will affect the prediction accuracy [18]. This paper adopts the gravitational search algorithm (GSA) [15] to optimize these hyperparameters, and automatically adjusts and optimizes the GRU network structure and training mode according to the input data to obtain the optimal combination of model parameters.

The GSA algorithm is a population optimization algorithm based on the law of universal gravitation and Newton's second law. As the algorithm loops, the particles

move continuously in the search space by the universal gravitation between them. When the particles move to the optimal position, the optimal solution is found. The specific method for the GSA algorithm to obtain the optimal parameters of the GRU network is as follows:

Step 1: Initializing the relevant parameters of the algorithm, determining the population size, the number of iterations and the change interval of the learning factor.

Step 2: Randomly generating a three-dimensional population particle (number of neurons, time step, batch size), and initializing the position and velocity of the particle. The dimension of the particle is the parameter that needs to be optimized.

Step 3: Calculating the fitness function value of each particle. The loss function of the GRU network is taken as the fitness function of the particle. The smaller the fitness function, the smaller the loss function of the model, and the better the combination of parameters obtained by the particle.

Step 4: Calculating the inertial mass of the particles, the gravity and acceleration of each particle in each direction.

Step 5: Updating the position and velocity of each particle according to the fitness value.

Step 6: When the number of iterations is reached or the fitness function of the particle becomes stable, it stops. The particle with the best position of the group is the optimal parameter combination obtained this time, otherwise, go to **Step 3** to continue the iteration.

3.3.2 GRU Network Correction Process

In view of the time-related characteristics of the network security situation, the specific procedure of correction method based on GRU network is as follows.

Step 1: The network security situation prediction value sequence $\hat{x}$ is input to the input end of the GRU network, and the training sample data is:

$$\hat{x}^{(0)} = (\hat{x}_1^{(0)}, \hat{x}_2^{(0)}, \cdots, \hat{x}_n^{(0)}).$$

Step 2: The parameters of the GRU neural network are initialized and set as the optimal combination obtained by the GSA algorithm. The training sample is input into the encoder GRU, and the input sequence $\hat{x}^{(0)}$ is mapped to the hidden state $\tilde{h}_t$ of the encoder through the GRU unit.

Step 3: Updating the hidden state of the GRU prediction network to h_t. ρ is a linear transformation function, and the hard_sigmoid function is selected in this paper. The residual value of network security situation prediction $e^{(0)}$ is taken as the output result, and the mean square error of all training samples is used as the target to train the GRU network.

$$e^{(0)}(t) = \rho(h_t, x^{(0)}(t)).$$

Step 4: The trained GRU network is used to predict the residual sequence, and the predicted residual value $\hat{e}_1^{(0)}$ was obtained.

Step 5: The accuracy of the preliminary prediction results is tested by a second revision. If the requirements are not met, return to **Step 2**; Otherwise, output the final network security situation prediction value: $\hat{y} = \hat{x}_1^{(0)} + \hat{e}_1^{(0)}$.

4 Experimental Simulation and Analysis

4.1 Experimental Dataset

In order to verify the performance of the proposed method in predicting the security situation of industrial control networks, two benchmark datasets are introduced in the model, which are the industrial control system intrusion detection dataset (Gas Pipeline dataset) published by Mississippi State University and the NSL-KDD dataset.

4.1.1 Gas Pipeline Dataset

The Gas Pipeline dataset is an intrusion detection data set of industrial control system published by Mississippi State University, which is derived from the network layer data of natural Gas Pipeline control system [1]. The dataset includes 1 type of normal data and 7 types of different attack data. Each record contains 26 traffic characteristics and category labels. The data description is shown in Table 1.

Table 1: Gas Pipeline data description

Category	Label	Description
Normal	0	Normal
NMRI	1	Simple malicious response injection attack
CMRI	2	Complex malicious response injection attack
MSCI	3	Malicious status command injection attack
MPCI	4	Malicious parameter command injection attack
MFCI	5	Malicious function command injection attack
DoS	6	Denial of service attack
RECO	7	Reconnaissance attack

Converting other characteristic data in the dataset into numerical data, and then preprocess these data. In order to ensure the authenticity of the situation prediction, this paper randomly selects 6000 consecutive samples for experiments.

4.1.2 NSL-KDD Dataset

The NSL-KDD dataset solves the inherent problems in the KDD99 dataset. The setting of NSL-KDD training set and test set is reasonable, and the evaluation results of different research work will be consistent and comparable [10]. NSL-KDD includes the training dataset KDDTrain+_20Percent and the test dataset KDDTest-21. The training dataset consists of 21 attack types, and 17 new attack types are added to the test set KDDTest-21. Each sample in the dataset is composed of 41-dimensional features and one-dimensional labels. The data description of NSL-KDD dataset is shown in Table 2.

The complete NSL-KDD dataset contains more than 140,000 records in total. If all of them are used as the verification dataset, it is too large. Therefore, 20% of the NSL-KDD dataset is selected as the sample data for experimental verification in this paper.

4.2 Experimental Parameter Setting

The dataset in Section 4.1 is divided into multiple time periods by hour. Based on Python3.6 and MATLAB, the network security situation assessment method based on text SimHash in [13] is adopted to carry out the situation assessment. The situation assessment value of each time slot is taken as the input of the prediction model to predict the network security situation at the next time point.

In Section 3.3.1, the GSA algorithm is used to optimize the GRU network parameters, and the GRU network structure and training mode are automatically adjusted and optimized according to the input data. The optimal combination of model parameters is obtained: the number of neurons is 5, the time step is 4, and the batch size is 1.

4.3 Selection of Evaluation Index

In the simulation experiment, two evaluation indexes are selected to evaluate the proposed prediction model, including mean absolute percentage error (MAPE) and root mean square error (RMSE). MAPE is a measure of the accuracy of the method of constructing fitting time series values in statistics, especially in trend estimation. A MAPE value of 0% represents a perfect model, while a value greater than 100% indicates an inferior model. RMSE is often used to measure the difference between model predicted values and actual observed values. The greater the difference, the greater the RMSE value. The specific formulas of MAPE and RMSE are defined as follows:

$$MAPE = \frac{100\%}{n}\sum_{i=1}^{n}\left|\frac{\hat{y}_i - y_i}{y_i}\right|$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2}$$

where, $\hat{y}_i$ and y_i respectively represent the predicted value and the actual value, and n represents the number of samples.

4.4 Analysis of Experimental Results

4.4.1 Comparison of Prediction Accuracy

In order to evaluate the prediction performance of each method as a whole, the final two errors of different models are calculated based on two datasets. The results are shown in Tables 3 and 4.

Table 3 shows the real network situation of 10 time periods based on the Gas Pipeline dataset, and successively shows the prediction results of the 6 network security situation prediction methods including this paper. In the process of simulation, the proposed method uses the situation value of the first four moments to predict the fifth moment, and the prediction effect is the best. It can be seen from Table 3 that the MAPE values of the prediction results of this method, LSTM [27] and Grey neural network [17] are 1.18%, 0.56%, and 10.94%, respectively. Obviously, the method based on machine learning has a better prediction effect on the fluctuating nonlinear time series. In addition, the MAPE values of SVM [14], BP neural network [24] and grey Verhulst model [11] are 26.82%, 21.07%, and 21.82%, respectively, indicating that the single-type prediction method could not effectively predict the network situation data of random volatility.

In order to show the changing trend of the network security situation more clearly, Figure 5 shows the network security situation prediction curves under different methods.

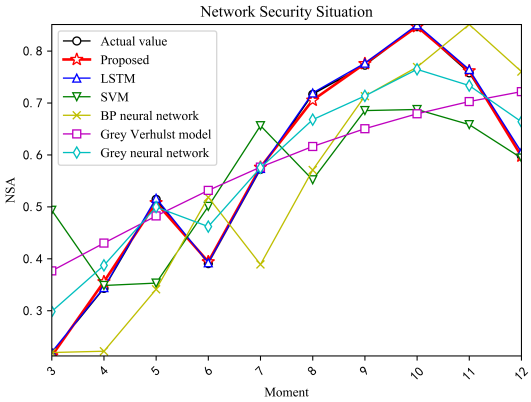


Figure 5: Comparison of situation prediction on Gas Pipeline Dataset

As can be seen from Figure 5 that the changes of predicted values and real values of several prediction methods are consistent. Obviously, the proposed method and LSTM network have the best fitting effect for the prediction of network security situation value. At moment 3, the predicted value of SVM method and grey Verhulst model

Table 2: NSL-KDD data description

Category	Label	Description
Normal	0	Normal
DoS	1	Denial of service attack
Probe	2	Surveillance and probing
R2L	3	Unauthorized access from a remote machine to a local machine
U2R	4	Unauthorized access to local superuser privileges by a local unprivileged user

Table 3: The prediction results of different models under the Gas Pipeline dataset

T/h	Actual	Proposed		References [27]		References [14]		References [24]		References [11]		References [17]	
		Pred	RE(%)	Pred	RE(%)	Pred	RE(%)	Pred	RE(%)	Pred	RE(%)	Pred	RE(%)
3	0.2189	0.2130	2.70	0.2203	0.64	0.4934	125.4	0.2195	0.27	0.3767	72.09	0.2985	36.36
4	0.3431	0.3558	3.70	0.3445	0.41	0.3486	1.60	0.2220	35.30	0.4302	25.39	0.3873	12.88
5	0.5147	0.5080	1.30	0.5167	0.39	0.3532	31.38	0.3412	33.71	0.4825	6.26	0.4996	2.93
6	0.3908	0.3942	0.87	0.3924	0.41	0.5009	28.17	0.5185	32.68	0.5317	36.05	0.4620	18.22
7	0.5723	0.5762	0.68	0.5740	0.30	0.6564	14.70	0.3891	32.01	0.5766	0.75	0.5753	0.52
8	0.7168	0.7054	1.60	0.7194	0.36	0.5521	22.98	0.5709	20.35	0.6163	14.02	0.6679	6.82
9	0.7729	0.7758	0.37	0.7767	0.49	0.6854	11.32	0.7132	7.72	0.6505	15.84	0.7136	7.67
10	0.8470	0.8471	0.01	0.8505	0.41	0.6875	18.83	0.7692	9.19	0.6792	19.81	0.7648	9.70
11	0.7581	0.7620	0.51	0.7645	0.84	0.6584	13.15	0.8512	12.28	0.7028	7.29	0.7336	3.23
12	0.5980	0.5976	0.07	0.6058	1.30	0.5940	0.67	0.7605	27.17	0.7219	20.72	0.6639	11.02
MAPE(%)		-	1.18	-	0.56	-	26.82	-	21.07	-	21.82	-	10.94
RMSE(%)		-	0.007	-	0.004	-	0.14	-	0.13	-	0.11	-	0.056

differs greatly from the actual value. The reason is that there are less data in front of the prediction period, which has little influence on the prediction period. Therefore, the method based on mathematical model is difficult to fit the relation between the data from the less time series data. In addition, the values of situation sequences are fluctuating, which is difficult to be described by specific functional expressions. After time 7, the prediction results of BP neural network and grey neural network are more consistent with the real value. As the amount of prediction data increases, the prediction method based on neural network has a better prediction effect on non-stationary random sequence, and the self-learning ability of the machine learning method is well reflected.

Table 4 shows the actual network situation of 12 time periods based on the NSL-KDD dataset, and successively shows the prediction results of the six network security situation prediction methods including this paper. It can be seen from the prediction results that the error of the test results of the six methods are different. The MAPE values of the prediction results of the proposed method, LSTM, BP neural network and grey neural network are 2.91%, 4.30%, 14.64%, and 14.88% respectively. It is obvious that the method based on machine learning has better predictive effect on the nonlinear time series with fluctuation. Moreover, the MAPE values of the prediction results of SVM and grey Verhulst model are 17.32% and 28.45%, successively, and the prediction effect is not ideal.

In order to show the changing trend of the network security situation more clearly, Figure 6 shows the network security situation prediction curves under different methods.

It can be seen from Figure 6 that the predicted values of several prediction methods are consistent with the actual values, but the proposed method and the LSTM network have the best prediction and fitting effect on the network security situation value. Compared with the Gas Pipeline dataset, selected NSL-KDD dataset corresponds to the network situation value sequence with less volatility, which can well explain the difference between MAPE and RMSE in Tables 3 and 4. According to Figure 5 and Figure 6, the grey Verhulst model is not ideal in predicting local volatility data. The proposed method makes use of the nonlinear processing capability of GRU to compensate for the problem of poor nonlinear fitting of the grey Verhulst model and achieves a good prediction effect.

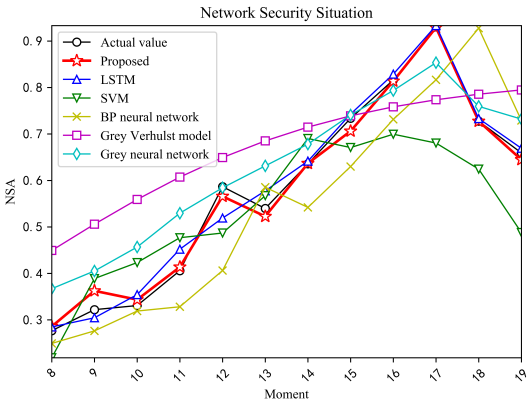


Figure 6: Comparison of situation prediction on NSL-KDD Dataset

4.4.2 Accuracy Test of Prediction Model

The prediction model can only be judged to be reasonable if it is tested, and only the tested model can be used for prediction. Common test indicators are: mean relative er-

Table 4: The prediction results of different models under the NSL-KDD dataset

T/h	Actual	Proposed		References [27]		References [14]		References [24]		References [11]		References [17]	
		Pred	RE(%)	Pred	RE(%)	Pred	RE(%)	Pred	RE(%)	Pred	RE(%)	Pred	RE(%)
8	0.2767	0.2858	3.29	0.2848	2.93	0.2185	21.03	0.2493	9.90	0.4493	62.38	0.3671	32.67
9	0.3220	0.3625	12.58	0.3045	5.43	0.3889	20.78	0.2763	14.19	0.5059	57.11	0.4052	25.84
10	0.3308	0.3435	3.84	0.3543	7.10	0.4233	27.96	0.3190	3.57	0.5591	69.01	0.4567	38.06
11	0.4058	0.4136	1.92	0.4520	11.38	0.4768	17.50	0.3282	19.12	0.6072	49.63	0.5296	30.51
12	0.5866	0.5658	3.55	0.5191	11.51	0.4869	17.00	0.4061	30.77	0.6493	10.69	0.5842	0.41
13	0.5397	0.5227	3.15	0.5776	7.02	0.5672	5.10	0.5854	8.47	0.6851	26.94	0.6313	16.97
14	0.6359	0.6367	1.26	0.6415	0.88	0.6903	8.55	0.5420	14.77	0.7148	12.41	0.6782	6.65
15	0.7337	0.7062	3.75	0.7429	1.25	0.6708	8.57	0.6300	14.13	0.7390	0.72	0.7409	0.98
16	0.8161	0.8136	0.31	0.8287	1.54	0.6995	14.29	0.7314	10.38	0.7584	7.07	0.7935	2.77
17	0.9306	0.9291	0.16	0.9332	0.28	0.6805	26.88	0.8166	12.25	0.7736	16.87	0.8534	8.30
18	0.7275	0.7274	0.01	0.7332	0.78	0.6244	14.17	0.9277	27.52	0.7856	7.99	0.7594	4.38
19	0.6593	0.6447	2.21	0.6692	1.50	0.4879	26.00	0.7290	10.57	0.7948	20.55	0.7320	11.03
MAPE(%)		-	2.91	-	4.30	-	17.32	-	14.64	-	28.45	-	14.88
RMSE(%)		-	0.017	-	0.028	-	0.11	-	0.10	-	0.14	-	0.076

ror, absolute correlation degree, ratio of mean square error and probability of small error. In general, the most frequently used indicators are mean relative error and mean square error ratio test. Given a set of time series values, a level of accuracy of the test model is determined. The commonly used classification of accuracy levels is shown in Table 5.

As can be seen from Tables 3 and 4 that the mean relative error value of grey Verhulst model is 21.82% and 28.45% in sequence. Combined with Table 5, it can be seen that the traditional grey Verhulst model is not applicable to network security situation prediction. The proposed method combines the improved adaptive grey Verhulst model with GRU network to improve the accuracy of prediction. From Tables 3 and 4, it can be seen that the mean relative error of the prediction results of the proposed method is 1.18% and 2.91% respectively, and the model accuracy is second level.

Assuming that $x^{(0)}$ is the original sequence, $\hat{x}^{(0)}$ is the corresponding prediction sequence, and $\varepsilon^{(0)}$ is the residual sequence, the mean square deviation of $x^{(0)}$ and $\varepsilon^{(0)}$ is S_1 and S_2 respectively.

$$C = \frac{S_2}{S_1}$$

For a given C_0 is greater than 0, when C is less than C_0, the prediction model is called the mean-variance ratio qualified model.

By combining the actual and predicted values in Tables 3 and 4, it can be calculated that the ratio of mean square error of the proposed method is 0.033 and 0.086 in turn. Therefore, the model accuracy is level 1. It can be seen that the model prediction accuracy is relatively high and can be utilized for network security situation prediction.

4.4.3 Analysis of Model Complexity

According to the description of model prediction steps in Section 3.2 and Section 3.3, the time complexity of each model training and prediction is analyzed. Assuming that N is the number of samples and m is the number of neurons, the number of internal iterations once trained by the model is $Q = m/batch$ size, and the model parameters are updated Q times iteratively. There are two gate structure functions and one candidate state in GRU unit, then the time complexity of GRU network is $O(Q \times 3 \times (N \times m + m^2))$. Furthermore, the time complexity of the initial prediction of time series by the adaptive grey Verhulst model is $O(N^2)$. Therefore, the time complexity of the proposed model is $O(Q \times 3 \times (N \times m + m^2) + N^2)$, compared with that of other prediction models, as shown in Table 6.

There are three gate functions and a candidate state inside the LSTM unit, so the time complexity of the method adopted in [27] is $O(Q \times 4 \times (N \times m + m^2))$. The SVM algorithm in the training stage is relatively complex, and its time complexity is related to many factors. The time complexity of standard SVM is $O(N^3)$. For a three-layer neural network, the time complexity of feed forward calculation mainly depends on two matrix operations. Assuming that the number of neurons in the input layer, the hidden layer and the output layer are m_1, m_2 and m_3 respectively, $m_1 \times m_2$ and $m_2 \times m_3$ calculations are required, so the time complexity of the method in [24] is $O(Q \times N \times m^2)$. The grey Verhulst model is a mathematical model, and its time complexity is mainly related to the process of data accumulation and reduction, so the time complexity is $O(N^2)$. Reference [17] combined grey theory and BP neural network to establish a grey neural network prediction method, with time complexity of $O(Q \times N \times m^2 + N^2)$, which is higher than the proposed method.

4.4.4 Aalysis of Convergence

Figure 7 shows how the training loss function values of different models change with the number of iterations on 20% of the NSL-KDD dataset. As SVM and grey Verhulst model differ from convergence analysis comparison the proposed method, no comparison is made.

It can be seen from Figure 7 that GRU, LSTM and BPNN reach the inflection point almost at the same time when conducting data training on 20% of the NSL-KDD

Table 5: Classification of prediction model accuracy

Level	Mean relative error	Absolute correlation degree	Ratio of mean square error	Probability of small error
1	0.01	0.90	0.35	0.95
2	0.05	0.80	0.50	0.80
3	0.10	0.70	0.65	0.70
4	0.20	0.60	0.80	0.60

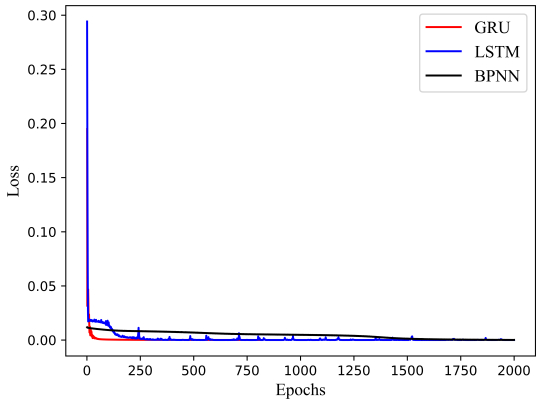


Figure 7: The relationship between the loss function value and the number of iterations

dataset, but GRU reaches the convergence state faster than LSTM and BPNN, and GRU loss value is smaller than the other two models, indicating that GRU network can learn data well.

Table 6: Comparison of time complexity

Prediction model	Time complexity
Proposed	$O(Q \times 3 \times (N \times m + m^2) + N^2)$
LSTM [27]	$O(Q \times 4 \times (N \times m + m^2))$
SVM [14]	$O(N^3)$
BP neural network [24]	$O(Q \times N \times m^2)$
Grey Verhulst model [11]	$O(N^2)$
Grey neural network [17]	$O(Q \times N \times m^2 + N^2)$

4.4.5 Analysis of Execution Time

The execution time referred to in this paper includes three parts: the initial prediction time of the adaptive grey Verhulst model, the time for GSA algorithm to solve the optimal parameter combination, the GRU network training time and the prediction time. The execution time of the method is closely related to the complexity of the model, iteration times of the algorithm, the quality of the hardware system and other factors. The running time of each model is recorded through experiments, as shown in Table 7.

As can be seen from Table 7 that the execution time of the proposed method is only better than that of the grey neural network, and this difference in execution time is mainly caused by the structure of the prediction process and prediction model. GRU has one less gate function than LSTM, so the number of parameters is less than LSTM. Therefore, the training speed of GRU is faster than that of LSTM on the whole. The LSTM execution time is shorter than the proposed method because of the process of parameter optimization. The SVM model and the grey Verhulst model have low execution time cost due to their simple processing. However, the prediction error is large and the network security situation value cannot be accurately predicted.

Table 7: Comparison of execution time

Prediction model	Execution time/s
Proposed	341.9314
LSTM [27]	246.8073
SVM [14]	165.5501
BP neural network [24]	23.3438
Grey Verhulst model [11]	0.3590
Grey neural network [17]	479.4507

5 Conclusions

A security situation prediction method for industrial control network based on adaptive grey Verhulst model and GRU network is presented. The proposed method solves the problems of the existing industrial control network security situation prediction methods, such as the insufficient accuracy of prediction, the difficulty of model construction, and the need of a large number of training data. The source of the background value error of the traditional grey Verhulst model is analyzed through the combination of number and shape, then the background value generation is reconstructed by the combination method of complex trapezoidal rule and complex Simpson's rule, and the improved grey Verhulst model is adopted to initially generate network security situation prediction sequence values. As a result, the trained GRU network is used to predict the residual and correct the security situation prediction value. The experimental results show that the prediction accuracy of the proposed method is higher than that of existing methods. In the field of industrial network security, the proposed method can help network administrators have more adequate time and preparation to deal with possible threats, reasonably allocate network resources, and take correct and effective defense measures against the network.

The shortcoming of the proposed method is that it is based on the influence of historical and current network situation values, and does not consider the factors that

affect the changes of network security situation. In the future work, forecasting the attack information, vulnerability information and service performance information in the future network will be the focus of research, and then comprehensively considering the three aspects of information to predict the complex situation changes of industrial control network more dynamically and reasonably.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (No. 61862041, 61363078). The authors also gratefully acknowledge the helpful comments and suggestions of the reviewers, which have improved the presentation.

References

[1] J. M. Beaver, R. C. Borges-Hink, M. A. Buckner, "An evaluation of machine learning methods to detect malicious scada communications," in *The 12th International Conference on Machine Learning and Applications*, vol. 2, pp. 54–59, Dec. 2013.

[2] G. Chen and Y. Q. Zhao, "Rf-svm based awareness algorithm in intelligent network security situation awareness system," in *The 3rd Workshop on Advanced Research and Technology in Industry (WARTIA'17)*, pp. 224–228, Sep. 2017.

[3] A. Erbad, R. Jain, K. Khan, D. Bhamare, M. Zolanvari and N. Meskin, "Cybersecurity for industrial control systems: A survey," *Computers & Security*, vol. 89, pp. 101677, 2020.

[4] C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk K. Cho, M. B. Van and Y Bengio, "Learning phrase representations using rnn encoder-decoder for statistical machine translation," *Computation and Language*, 2014. arXiv:1406.1078.

[5] Y. Z. Hang, X. Lv, X. L. Han, Y. Liu and Y. Li, "Network security situation prediction method based on ifs-narx model," *Journal of Jilin University (Engineering and Technology Edition)*, vol. 49, no. 2, pp. 592–598, 2019.

[6] C. R. He and J. Zhu, "Security situation prediction method of gru neural network based on attention mechanism," *System Engineering and Electronic Technology*, vol. 43, pp. 264–272, 2021.

[7] X. Y. Hong, "Network security situation prediction based on grey relational analysis and support vector machine algorithm," *International Journal of Network Security*, vol. 1, pp. 177–182, 2020.

[8] Q. Hu, M. R. Asghar, and S. Zeadally, "Cybersecurity in industrial control systems: Issues, technologies, and challenges," *Computer Networks*, vol. 165, pp. 106946, 2019.

[9] G. Q. Huang, X. Zhao, B. Zhang, Y. Li, C. Z. Wang and Y. C. Li, "Analysis framework of network security situational awareness and comparison of implementation methods," *EURASIP Journal on Wireless Communications and Networking*, vol. 2019, no. 1, pp. 1–32, 2019.

[10] L. Journaux, Y. Hamid, V. R. Balasaraswathi and M. Sugumaran, "Benchmark datasets for network intrusion detection: A review," *International Journal Network Security*, vol. 20, no. 4, pp. 645–654, 2018.

[11] Y. B. Leau and S. Manickam, "A novel adaptive grey verhulst model for network security situation prediction," *International Journal of Advanced Computer Science & Applications*, vol. 1, no. 7, pp. 90–95, 2016.

[12] P. W. Lin and Y. Chen, "Dynamic network security situation prediction based on bayesian attack graph and big data," in *IEEE 4th Information Technology and Mechatronics Engineering Conference (ITOEC'18)*, pp. 176–180, Dec. 2018.

[13] P. W. Lin and Y. H. Chen, "Network security situation assessment based on text simhash in big data environment.," *International Journal Network Security*, vol. 21, no. 4, pp. 699–708, 2019.

[14] C. Liu, Z. Y. Shi, H. Z. Yan, J. J. Hu, D. Y. Ma and C. Z. Hu, "Network security situation prediction based on mr-svm," *IEEE Access*, vol. 7, pp. 130937–130945, 2019.

[15] H. Nezamabadi-Pour, E. Rashedi and S. Saryazdi, "GSA: A gravitational search algorithm," *Information Sciences*, vol. 179, no. 13, pp. 2232–2248, 2009.

[16] L. Shen and Z. C. Wen, "Network security situation prediction in the cloud environment based on grey neural network," *Journal of Computational Methods in Sciences and Engineering*, vol. 19, no. 1, pp. 153–167, 2019.

[17] Y. J. Shen, S. B. Zhang and G. D. Zhang, "Network security situation prediction model based on multi-swarm chaotic particle optimization and optimized grey neural network," in *IEEE 9th International Conference on Software Engineering and Service Science (ICSESS'18)*, pp. 426–429, Nov. 2018.

[18] Y. Singh, D. Srivastava and A. Sahoo, "Auto tuning of rnn hyper-parameters using cuckoo search algorithm," in *The Twelfth International Conference on Contemporary Computing*, pp. 1–5, Aug. 2019.

[19] M. Y. Tong, B. Zeng and X. Ma, "A new-structure grey verhulst model: Development and performance comparison," *Applied Mathematical Modelling*, vol. 81, pp. 522–537, 2020.

[20] Y. Q. Tu, X. D. Kang, W. P. Chen, Z. G. Ao and Z. N. Zhao, "Network situation awareness model prediction method based on genetic optimization support vector machine," in *International Conference on Computer Networks and Communication Technology (CNCT'16)*, pp. 493–500, Dec. 2016.

[21] B. Wang, W. Kong, H. Guan, and N. N. Xiong, "Air quality forecasting based on gated recurrent long short term memory model in internet of things," *IEEE Access*, vol. 7, pp. 69524–69534, 2019.

[22] S. W. Wang, X. Y. Zhang, J. Z. Lu, G. S. Dong, W. C. Li and X. Li, "The assessment method of network security situation based on improved bp neural

network," in *International Conference on Computer Engineering and Networks*, pp. 67–76, Aug. 2018.

[23] X. H. Wei, Y. Jiang, C. H. Li and Z. P. Li, "Research on network security situation prediction of rbf with improved pso optimization," *Measurement and Control Technology*, vol. 315, no. 5, pp. 63–67, 2018.

[24] L. Y. Yan, C. Z. Wang, X. Zhang, Z. W. Ye and R. X. Wang, "Security situation prediction based on hybrid rice optimization algorithm and back propagation neural network," in *IEEE 4th International Symposium on Wireless Systems within the International Conferences on Intelligent Data Acquisition and Advanced Computing Systems (IDAACS-SWS'18)*, pp. 73–77, Sep. 2018.

[25] H. Y. Yang and X. G. Zhang, "Self-corrected coefficient smoothing method based network security situation prediction," *Journal on Communications*, vol. 41, no. 5, pp. 196–204, 2020.

[26] X. D. Yan, X. D. Zheng, W. Liang, Z. Chen and P. Zhuo, "Multiscale entropy-based weighted hidden markov network security situation prediction model," in *IEEE International Congress on Internet of Things (ICIOT'17)*, pp. 97–104, June 2017.

[27] J. J. Zhang, Q. Fu, Q. Zhao, L. Shang, W. Zhao and Y. Yang, "Network security situation prediction based on long short-term memory network," in *The 20th Asia-Pacific Network Operations and Management Symposium (APNOMS'19)*, pp. 1–4, Sep. 2019.

Biography

Rui-hong Dong. Researcher, worked at school of computer and communication in Lanzhou university of technology. His research interests include network and information security, information hiding and steganalysis analysis, computer network.

Chuang Shu. Received the BS degree in network engineering from Hubei University of Technology, Wuhan, China, in 2018. Currently, he is studying for a master's degree in Lanzhou University of Technology. His research interests include network and information security, industrial control network security.

Zhang Qiu-yu. Researcher/Ph.D. supervisor, graduated from Gansu University of Technology in 1986, and then worked at school of computer and communication in Lanzhou University of Technology. He is vice dean of Gansu manufacturing information engineering research center, a CCF senior member, a member of IEEE and ACM. His research interests include network and information security, information hiding and steganalysis, multimedia communication technology.

Ya-Yu Mo. He is currently studying for a BS degree in Lanzhou University of Technology, Lanzhou, China. His research interests include Software Theory, Game Engineering and Computational Intelligence.

A New Upper Bound and Optimal Constructions of Equi-Difference Conflict-Avoiding Codes on Constant Weight

Chun-E Zhao and Yuhua Sun
(Corresponding author: Chun-e Zhao)
College of Sciences, China University of Petroleum
Shandong Qingdao, 266555, China
(Email: zhaochune1981@163.com)
(Received Dec. 27, 2020; Revised and Accepted July 5, 2021; First Online Nov. 9, 2021)

Abstract

Conflict-avoiding codes (CACs) have been used in multiple-access collision channels without feedback. The size of a CAC is the number of potential users that can be supported in the system. A code with a maximum size is called optimal. An optimal CAC enables the largest possible number of asynchronous users to transmit information efficiently and reliably. In this paper, a new upper bound on the maximum size of arbitrary Equi-difference CAC is presented. Furthermore, three optimal constructions of Equi-difference CACs are also given. One is a generalized construction for prime length $L = p$, and the other two are for two-prime length $L = pq$.

Keywords: Conflict-Avoiding Codes; Equi-Difference Codes; Exceptional Code; Non Exceptional Code; Optimal Construction

1 Introduction

Nowadays communication has become an indispensable part of people's daily life. Coding plays an important role in kinds of communication systems, especially in multi-access communication system. Multi-access channels are widely used in the fields of mobile and satellite communication networks. TDMA (time-division multiple address) is an important multi-access technique.

In a TDMA system, the satellite working time is divided into periodic frames and each frame is then divided into some time slots. In order to support user-irrepressibility, each user is assigned a protocol sequence which is derived by a CAC codeword. So conflict-avoiding codes have been studied as protocol [2] sequences for a multiple-access channel (collision channel) without feedback [4, 5, 12]. And the technical description of such a multiple-access channel model can be found in [6]. The protocol sequence is a binary sequence and the number of ones in it is called its Hamming weight. The Hamming weight k of the sequence is the minimum weight requirement for user-irrepressibility and it also means the maximal number of active users who can send packets in the same time slot.

For fixed length L, many works are devoted to determine the maximal number of potential users for Hamming weight three in [1, 3, 4, 7, 8, 15]. Some optimal constructions for constant weight are presented in [14] and [9]. An asymptotic version of this general upper bound can be found in [10]. A general upper bound on the number of potential users for all Hamming weights is provided in [13].

In [10], the asymptotic bound on the size of constant-weight conflict-avoiding codes have been discussed. This is about the arbitrarily CACs, so the bound is not the best for special cases. In this paper, we will focus on the equi-difference CACs. First, an upper bound on the maximum size of equi-difference CACs for constant weight is presented. This upper bound is lower than the former ones and its concise expression will greatly reduce the time complexity of validation. Secondly, three new constructions of optimal equi-difference CACs are presented. Correspondingly, the range of CACs constructed will be enlarged. Third, the results show that these new codewords' size can reach this new upper bound. As a result, these CACs constructed are optimal and this new upper bound can be reached.

2 Preliminaries

Let $P(L, w)$ be the set of all w-subsets of $Z_L = \{0, 1, ..., L-1\}$ and $Z_L^* = Z_L \setminus \{0\}$. Given a w-subset $I \in P(L, w)$, we define the *set of difference* of I by

$$d^*(I) = \{j - i | i, j \in I, i \neq j\},$$

where the $j - i$ is modulo L. A *conflict-avoiding code* (CAC) C of length L and weight w is a subset $C \subset P(L, w)$ satisfying the following condition

$$d^*(I_j) \bigcap d^*(I_k) = \emptyset \text{ for any } I_j, I_k \in C, j \neq k.$$

Each element $I \in C$ is called a *codeword* of length L and weight w.

A codeword I is called equi-difference if the elements in I has the following form $I = \{0, g, 2g, ..., (w-1)g\}$ for some $g \in Z_L$, where the product jg is reduced mod L, for $j = 0, 1, 2, 3, ..., w-1$. The element g is called a generator of this codeword. For an equi-difference codeword I generated by g, the set of difference is

$$d^*(I) = \{\pm g, \pm 2g, ..., \pm(w-1)g\}.$$

We note that the elements $\pm g, \pm 2g, ..., \pm(w-1)g$ may not be distinct mod L. Hence in general we have $|d^*(I)| \leq 2w-2$. We adopt the terminology in [9] and say that a codeword I of weight w is *exceptional* if $|d^*(I)| < 2w-2$. Let $d(I) = d^*(I) \cup \{0\}$, then $|d^*(I)| < 2w-2$ is equivalent to $|d(I)| < 2w-1$.

If all codewords in a CAC C are equi-difference, then C is called equi-difference. Let $CAC^e(L, w)$ denote the class of all equi-difference CACs of length L and weight w. For every code $C \in CAC^e(L, w)$, $C = C_1 \cup C_2$ always holds, where $C_1 = \{I \in C, I \text{ is exceptional}\}$ and $C_2 = \{I \in C, I \text{ is non exceptional}\}$. The maximal size of $CAC^e(L, w)$ is denoted by $M^e(L, w)$, *i.e.*

$$M^e(L, w) = \max\{|C||C \in CAC^e(L, w)\}.$$

A code $C \in CAC^e(L, w)$ is called optimal if $|C| = M^e(L, w)$. An optimal code $C \in CAC^e(L, w)$ is called tight if $\bigcup_{I \in C} d^*(I) = Z_L^*$.

3 An Upper Bound on Equi-Difference CACs

Lemma 1. *[10]: $|d(A)| \geq |A|$ for any subset A in G.*

Lemma 2. *[10]: Let $w(n)$ denote the number of distinct prime divisors of n. For $n \geq 2$ and $w \geq 2$, we have*

$$M(n, w) \leq \frac{n-1}{2w-2} + \frac{w(n)}{2}$$

Lemma 3. *[13]: For $L \geq w \geq 2$,*

$$M(L, w) \leq \lfloor \frac{L-1+F(L,w)}{2w-2} \rfloor,$$

where $F(L, w) := \max_{S \in \ell(L,w)} \sum_{x \in S} (x - 1 - 2x\lceil w/x \rceil + 2w)$, $\ell(L, w) := \{S \subseteq S(L, w) : \gcd(i, j) = 1, \forall i, j \in S, i \neq j\}$, $S(L, w) := \{x \in \{2, 3, \cdots, 2w-2\} : x \text{ divides } L, \text{ and } 2x\lceil w/x \rceil - x \leq 2w-2\}$.

There are two upper bounds listed above and they have their own merits and drawbacks. In this paper, we give a new upper bound for equi-difference CACs using a similar method to the one in [11]. And it is easier to be reached than the one in Lemma 2 and more easily to deal with problems than the one in Lemma 3.

Theorem 1. *Let $\Omega(L, w) = \{p \text{ is a divisor of } L \text{ and } w \leq p < 2w-1\}$. For $n \geq 2$ and $w \geq 2$, then*

$$M^e(L, w) \leq \frac{L-1+\sum_{p \in \Omega^*(L,w)} (2w-1-p)}{2w-2},$$

where $\Omega^*(L, w) = \{p \in \Omega(L, w) | p \text{ is prime or } p \text{ satisfies if } \gcd(p, p') \neq 1 \text{ for } p' \in \Omega(L, w), p \leq p' \text{ always holds}\}$.

Proof. Let C be an (L, w)-equi-difference CAC, in which there are E exceptional codewords. Suppose $C = C_1 \cup C_2$, where $C_1 = \{I \in C | I \text{ is non exceptional}\}$ and $C_2 = \{I \in C | I \text{ is exceptional}\}$. Then $|C_2| = E$. For $i = 1, 2, \cdots, E$, denote the *ith* exceptional codeword by I_i and let $|d^*(I_i)| = f_i$. Then we have the following inequalities:

$$\begin{aligned} (2w-2)|C_1| + \sum_{i=1}^{E} f_i &\leq L-1. \\ (2w-2)(|C_1|+E) &\leq L-1+\sum_{i=1}^{E}(2w-2-f_i). \end{aligned}$$

In fact, for every exceptional codeword I_i with generator g_i, $d(I_i)$ is a subgroup with generator $\gcd(L, g_i)$ and then $|d(I_i)|$ is a divisor of L. Let $|d(I_i)| = p_i$, then

$$(2w-2)(|C_1|+E) \leq L-1+\sum_{i=1}^{E}(2w-1-p_i). \quad (1)$$

For every two exceptional codewords I_1, I_2, $d^*(I_1) \bigcap d^*(I_2) = \emptyset$ if and only if their generators are relatively prime. This implies that $\gcd(|d(I_1)|, |d(I_2)|) = 1$. So each element in $\{p_1, p_2, \cdots, p_E\}$ satisfies: (i) $p_i | L$; (ii) $\gcd(p_i, p_j) = 1$, for $i \neq j$; and (iii) $w \leq p_i < 2w-1$.

So $\sum_{i=1}^{E}(2w-1-p_i) \leq \sum_{p \in \Omega^*(L,w)} (2w-1-p)$, $|C| = |C_1| + E$, then Equation (1) turns to be

$$|C| \leq \frac{L-1+\sum_{p \in \Omega^*(L,w)} (2w-1-p)}{2w-2} \quad (2)$$

So we have

$$M^e(L, w) \leq \frac{L-1+\sum_{p \in \Omega^*(L,w)} (2w-1-p)}{2w-2} \quad (3)$$

□

Using this bound, we can easily deal with the Theorems 5-8 in [10] and Corollary 7 in [13] for equi-difference condition. And we also get the following Corollary 1 immediately.

Corollary 1. *Let L be an integer factorized as $2^a 3^b 5^c 7^d l$, where l is not divisible by $2, 3, 5$ or 7. Then we have*

$$M^e(L, w) \leq \begin{cases} \lfloor \frac{L+2}{4} \rfloor, & \text{for } w = 3; \\ \lfloor \frac{L+4}{6} \rfloor, & \text{for } w = 4; \\ \lfloor \frac{L+8}{8} \rfloor, & \text{for } w = 5; \\ \lfloor \frac{L+8}{10} \rfloor, & \text{for } w = 6. \end{cases}$$

4 Constructions on Optimal Equi-Difference CACs

We adopt the notation in [9] in the following. For a subgroup H of G with $\frac{|G|}{|H|} = f$, if each coset H_j of H contains exactly one element in $\{i_1, i_2, \cdots, i_f\}$ for $j = 1, 2, \cdots, f$, then $\{i_1, i_2, \cdots, i_f\}$ is said to form a system of distinct representatives(SDR for short) of $\{H_1, H_2, \cdots, H_f\}$. Let $Z_L^{\times} = \{a \in Z_L | gcd(a, L) = 1\}$.

Condition 1. There exists a subgroup H of $Z_L^{\times}$ such that $-1 \in H$, $|H| = \frac{|Z_L^{\times}|}{(w-1)}$ and $\{1, 2, \cdots, w-1\}$ forms a SDR of H's cosets.

Condition 2. There exists a subgroup H of $Z_L^{\times}$ such that $-1 \in H$, $|H| = \frac{|Z_L^{\times}|}{2(w-1)}$ and $\{\pm 1, \pm 2, \cdots, \pm(w-1)\}$ forms a SDR of H's cosets.

4.1 Optimal Construction on Equi-Difference CACs of Length $L = p$

Lemma 4. *[9]: Let $p = 2(w-1)m + 1$ be a prime number and suppose that $\{1, 2, \cdots, w-1\}$ forms a SDR of $\{H_j^{w-1}(p) : j = 0, \cdots, w-2\}$. Let α be a primitive element in the finite field Z_p and let $g = \alpha^{w-1}$. Then the m codewords of weight w generated by $1, g, g^2, \cdots, g^{m-1}$ form an equi-difference $(2(w-1)m+1, w) - CAC$.*

The p's satisfying SDR in Lemma 4 are rare. Following, we will give a generalized construction in which the range of p's will be enlarged.

Theorem 2. *Let $p = 2(w-1)ms + 1$ be a prime number and H be a subgroup of Z_p^* with order $2m(w-1)$. Suppose that $\{1, 2, \cdots, w-1\}$ forms a SDR of $\{N_1, N_2, \cdots, N_{w-1}\}$, where N_1 is a subgroup of H with order $2m$ and N_j's are the cosets of N_1 in H. Let α be a primitive element in the finite field Z_p and let $g_{ij} = \alpha^{i+s(w-1)j}$ for $0 \le i \le s-1, 0 \le j \le m-1$. Then the sm codewords of weight w generated by g_{ij} form an optimal equi-difference $(2(w-1)ms+1, w) - CAC$.*

Proof. Because H is a subgroup of Z_p^* with order $2m(w-1)$, then there exists an primitive element α of Z_p^* such that $H = (\alpha^s)$. Then

$$Z_p^* = \bigcup_{i=0}^{s-1} \alpha^i H.$$

Let $g = \alpha^{s(w-1)}$ and $N_1 = (g)$ be the subgroup of H with order $2m$. For $\{1, 2, \cdots, w-1\}$ forms a SDR of the cosets of N_1, then

$$H = \bigcup_{j=1}^{w-1} jN_1$$

And for the the reason that the order of N_1 is $2m$ and $g^m = \alpha^{s(w-1)m} = \alpha^{\frac{p-1}{2}} = -1$, so $N_1 = \{\pm 1, \pm g, \pm g^2, \cdots, \pm g^{m-1}\}$. Let $A = \{\pm 1, \pm 2, \cdots, \pm(w-1)\}$, then $H = \cup_{j=1}^{w-1} jN_1 = \cup_{t=0}^{m-1} g^t A$.

So

$$Z_p^* = \bigcup_{i=0}^{s-1} \alpha^i H = \bigcup_{i=0}^{s-1} \bigcup_{j=0}^{m-1} \alpha^i g^j A.$$

Let $\Gamma(C) = \{\alpha^i g^j, 0 \le i \le s-1, 0 \le j \le m-1\}$ be the set of generators of C, then $I_{(i,j)} = \{0, \alpha^i g^i, 2\alpha^i g^i, \cdots, (w-1)\alpha^i g^i\}$, for $0 \le i \le s-1, 0 \le j \le m-1$. And

$$d^*(I_{(i,j)}) \cap d^*(I_{(k,s)}) = \emptyset$$

for $(i, j) \ne (k, s)$. So $C = \{I_{(i,j)} | 0 \le i \le s-1, 0 \le j \le m-1\}$ forms an equi-difference CAC. The size of C is

$$|C| = sm = \frac{2m(w-1)s}{2w-2} = \frac{p-1}{2w-2}$$

By Theorem 1, C is an optimal equi-difference CAC. And

$$Z_p^* = \bigcup_{i=0}^{s-1} \bigcup_{j=0}^{w-2} d^*(I_{(i,j)}).$$

So C is an optimal and tight equi-difference CAC. □

For $s = 1$ in Theorem 1, it is exactly the construction mentioned in Lemma 4. So it is a generalized construction. And the following Example1 shows that it is a real generalization.

Example 1. *Let $p = 919$, and in the expression $p = 2(w-1)ms+1$, let $w = 4, m = 51, s = 3$ be the parameters and $\alpha = 7$ the primitive element of Z_p^*. Let $N_1 = (\alpha^9)$ and $H = (\alpha^3)$ be the subgroup generated by α^9 and α^3, respectively. N_1 is a subgroup of H. We can check that $\{1, 2, 3\}$ forms a SDR of the cosets N_1, N_2, N_3 in H. The 153 codewords generated by the generators form an optimal (919,4)-CAC C. The set of the generators is*

$\Gamma(C) = \{1, 7, 49, 317, 381, 388, 878, 635, 769, 788, 34, 238, 747, 669, 88, 616, 703, 326, 444, 453, 414, 141, 237, 740, 585, 690, 235, 726, 8, 56, 392, 698, 291, 199, 706, 347, 591, 485, 638, 790, 272, 66, 462, 757, 704, 333, 110, 770, 795, 867, 555, 209, 58, 406, 85, 642, 294, 64, 448, 379, 70, 490, 673, 134, 19, 133, 204, 509, 806, 338, 528, 20, 542, 118, 826, 880, 646, 846, 503, 764, 753, 464, 491, 680, 48, 336, 514, 512, 827, 275, 560, 244, 789, 153, 152, 145, 713, 396, 15, 866, 548, 160, 660, 25, 175, 607, 573, 335, 348, 598, 510, 36, 252, 845, 384, 850, 436, 420, 183, 362, 804, 114, 798, 305, 297, 241, 190, 411, 120, 495, 708, 361, 685, 200, 481, 261, 908, 842, 27, 189, 404, 288, 178, 327, 315, 367, 731, 603, 545, 139\}$.

4.2 Optimal Constructions on Equi-Difference CACs of Length $L = pq$

After we have constructed CACs for prime length $L = p$ based on Theorem 2, we will give a recursive construction of CACs for two-prime length $L = pq$.

Theorem 3. *Let C_1 be an optimal tight (p_1, w)-equi-difference CAC with m_1 codewords and C_2 be an optimal tight (p_2, w)-equi-difference CAC with m_2 codewords. Then the set*

$$C = \{I_{(k,i)}, J_j^{'} | 1 \le k \le m_1, 0 \le i \le p_2 - 1, 1 \le j \le m_2\}$$

forms a (p_1p_2, w) optimal equi-difference CAC with $m_1p_2 + m_2$ codewords, where $I_{(k,i)} = (0, a_1 + ip_1, a_2 + 2ip_1 \cdots, a_{w-1} + (w-1)ip_1)$ for $I_k = (0, a_1, a_2, \cdots, a_{w-1}) \in C_1$, $J_j^{'} = (0, b_1p_1, b_2p_1, \cdots, b_{w-1}p_1)$ for $J_j = (0, b_1, b_2, \cdots, b_{w-1}) \in C_2$.

Proof.

1) $d^*(I_{(k_1,i_1)}) \cap d^*(I_{(k_2,i_2)}) = \emptyset$, $(k_1, i_1) \neq (k_2, i_2)$.

Let $I_{(k,i)} = (0, g_k + ip_1, 2g_k + 2ip_1, \cdots, (w-1)g_k + (w-1)ip_1)$ in C for $I_k = (0, g_k, 2g_k, \cdots, (w-1)g_k) \in C_1$. Then $d^*(I_{(k,i)}) = \{\pm(g_k + ip_1), \pm(2g_k + 2ip_1), \cdots, \pm((w-1)g_k + (w-1)ip_1)\}$. For $(k_1, i_1) \neq (k_2, i_2)$, if there exists some $xg_{k_1} + xi_1p_1 = yg_{k_2} + yi_2p_1$ for $-(w-1) \le x, y \le (w-1)$, then

$$xg_{k_1} - yg_{k_2} = (yi_2 - xi_1)p_1 \pmod{p_1p_2}. \quad (4)$$

So $xg_{k_1} = yg_{k_2} \pmod{p_1}$. Then $x = y, g_{k_1} = g_{k_2}$ for $xg_{k_1} = yg_{k_2} \in Zp_1^*$ and $xg_{k_1} \in d^*(I_{k_1}), yg_{k_2} \in d^*(I_{k_2})$. So $k_1 = k_2$. And in Equation (4), we have $yi_2 = xi_1 \pmod{p_1p_2}$. So $x = y$ and $i_1 = i_2$ for $-(w-1) \le x, y \le (w-1)$ and $0 \le i_1, i_2 \le p_2 - 1$ which contracts with $(k_1, i_1) \neq (k_2, i_2)$. So $d^*(I_{(k_1,i_1)}) \cap d^*(I_{(k_2,i_2)}) = \emptyset$ for $(k_1, i_1) \neq (k_2, i_2)$.

2) $d^*(J_{j_1}^{'}) \cap d^*(J_{j_2}^{'}) = \emptyset$, $j_1 \neq j_2$.

Let $J_j^{'} = (0, b_jp_1, 2b_jp_1, \cdots, (w-1)b_jp_1)$ for $J_j = (0, b_j, 2b_j, \cdots, (w-1)b_j) \in C_2$. Then $d^*(J_j^{'}) = \{\pm b_1p_1, \pm 2b_1p_1, \cdots, \pm(w-1)b_1p_1\}$. For $j_1 \neq j_2$, if there exists $x, y \in \{-(w-1), (w-1)\}$, such that

$$xb_{j_1}p_1 = yb_{j_2}p_1 \pmod{p_1p_2}.$$

Then $xb_{j_1} - yb_{j_2} = 0 \pmod{p_2}$ also holds. So $xb_{j_1} = yb_{j_2} \in d^*(J_{j_1}) \cap d^*(J_{j_2})$ which contracts with $d^*(J_{j_1}) \cap d^*(J_{j_2}) = \emptyset$. So $d^*(J_{j_1}^{'}) \cap d^*(J_{j_2}^{'}) = \emptyset$ for $j_1 \neq j_2$.

3) $d^*(I_{(k,i)}) \cap d^*(J_j^{'}) = \emptyset$, for any k, i, j.

If there exists some $x(g_k + ip_1) = y(b_jp_1 \pmod{p_1p_2})$, then $xg_k = 0 \pmod{p_1}$ which contracts with $xg_k \in d^*(I_k)$. So $d^*(I_{(k,i)}) \cap d^*(J_{(j,j)}) = \emptyset$. By (1)(2)(3), C is an equi-difference CAC.

4) $|C| = m_1p_2 + m_2$.

On the other aspect, by Lemma 1 $|C| \le \lfloor \frac{L-1}{2w-1} \rfloor = \frac{p_1p_2-1}{2w-2} = \frac{p_2(p_1-1)+p_2-1}{2w-2} = \frac{p_1-1}{2w-2}p_2 + \frac{p_2-1}{2w-2} = m_1p_2 + m_2$.

So C is an optimal equi-difference CAC. □

Furthermore, we will give another optimal construction of CACs for two prime length $L = pq$. It indicates that the new upper bound given in Theorem 1 is lower than the known one listed in Lemma 2 and is more convenient than the one listed in Lemma 3.

Theorem 4. *Let $L = pq$, where $q = 2(w-1)f + 1$ and $w \le p \le 2(w-1)$ are both primes. If L and q satisfy condition 1 or condition 2, then there exists an optimal equi-difference $(L, w) - CAC$ C with $|C| = pf + 1$.*

Proof. It is well known that $Z_L^{\times}$ is a multiplicative group. Let $(p) = \{kp \pmod{L} | k \in Z\}$ and $(q) = \{kq \pmod{L} | k \in Z\}$ be the additive subgroups of Z_L. It is clear that $Z_L^* = Z_L^{\times} \cup (p)^* \cup (q)^*$, where $(p)^* = (p) \setminus \{0\}$, $(q)^* = (q) \setminus \{0\}$. Take L, q satisfying condition 1 for example. We consider the elements in $Z_L^{\times}$, $(p)^*$ and $(q)^*$, respectively.

1) Elements in $Z_L^{\times}$.

Because L satisfies condition1, there exists a subgroup H of $Z_L^{\times}$ such that $-1 \in H$ and $\{1, 2, \cdots, (w-1)\}$ forms a SDR of the cosets. Denoted $|H|$ by $2s$. Let α be the generator of H. Select each $g_i = \alpha^i \in H$ as the generator of $I_i = \{0, g_i, 2g_i, \cdots, (w-1)g_i\}, i = 1, 2, \cdots, s$, then $d^*(I_i) = \{\pm g_i, \pm 2g_i, \cdots, \pm(w-1)g_i\}$. If $i \neq j, 1 \le i, j \le s$, then $d^*(I_i) \cap d^*(I_j) = \phi$ and $\cup_{i=1}^{s} d^*(I_i) = Z_L^{\times}$.

2) Elements in $(p)^*$.

Because $|(p)^*| = |Z_q^*|$ and q satisfies condition 1, then there exists a subgroup N of Z_q^* such that $-1 \in N$ and $\{1, 2, \cdots, (w-1)\}$ forms a SDR of the cosets of N. Denoted $|N|$ by $2t$. Let β be the generator of N. Select each $b_i = \beta^i p \pmod{L} \in (p)^*$ as the generator to construct a codeword $J_i = \{0, b_i, 2b_i, \cdots, (w-1)b_i\}, i = 1, 2, \cdots, t$, then $d^*(J_i) = \{\pm b_i, \pm 2b_i, \cdots, \pm(w-1)b_i\}$. If $i \neq j, 1 \le i, j \le t$, then $d^*(J_i) \cap d^*(J_j) = \phi$ and $\cup_{i=1}^{t} d^*(I_i) = Z_q^*$.

3) Elements in $(q)^*$.

If the codeword generated by q is noted by K, then $|d^*(K)| = |(q)^*| = p - 1 \le 2(w-1) - 1 < 2(w-1)$, so K is exceptional and $(q)^* = d^*(K)$. Then $C = \{I_1, I_2, \cdots, I_s, J_1, J_2, \cdots, J_t, K\}$ forms an equi-difference conflict-avoiding code. The size of C is

$$|C| = s+t+1 = \frac{(p-1)(q-1)}{2(w-1)} + \frac{q-1}{2(w-1)} + 1 = pf+1$$

On the other hand, by Theorem 1, the size of the code satisfies

$$\begin{aligned} |C| &\le \frac{L - 1 + \sum\limits_{p \in \Omega^*(L,w)} (2w - 1 - p)}{2w - 2} \\ &= \frac{L - 1 + (2w - 1 - p)}{2w - 2} \\ &= \frac{L - p}{2w - 2} + 1 = \frac{p(2(w-1))f}{2w - 2} + 1 = pf + 1 \end{aligned} \quad (5)$$

So this construction is optimal. □

Example 2. *Let* $L = 671, w = 11, p = 11, q = 61, f = 3$ *be the parameters. We can check that* $H = (45)$ *is a subgroup satisfies condition 2, and* $s = 30, g = 45$. *The 34 codewords generated by the generators form an optimal (671,11)-CAC C. The set of the generators is* $\Gamma(C) = \{1, 45, 12, 540, 144, 441, 386, 595, 606, 430, 562, 463, 34, 188, 408, 243, 199, 232, 375, 100, 474, 529, 320, 309, 485, 353, 452, 210, 56, 507, 11, 121, 231, 61\}$.

5 Conclusion

In this paper, we first give a new upper bound of equi-difference CACs. Using this bound, it is easier to be reached and is easier to deal with some problems. Secondly, we give three optimal constructions of equi-difference CACs. One shows the superiority of the new upper bound and the other two make the range of optimal CACs constructed enlarged.

Acknowledgments

The work is financially supported by the National Natural Science Foundation of China (No. 61902429, No.11775306), the Fundamental Research Funds for the Central Universities (No. 19CX02058A), Shandong Provincial Natural Science Foundation of China (ZR2019MF070).

References

[1] H. L. Fu, M. Mishima and S. Uruno, "Optimal conflict avoiding codes of length n=(0 mod16) and weight 3," *Designs, Codes and Cryptography*, vol. 52, no. 3, pp. 275–291, 2009.

[2] X. He, Z. Y. Ji, W. R. Liu, "An improved authentication protocol for telecare medical information system," *International Journal of Electronics and Information Engineering*, vol. 12, no. 4, pp. 170–181, 2020.

[3] S. Janiszewski, A.Y. Teymorian, M. Jimbo, M. Mishima and V.D. Tonchev, "On conflict avoiding codes of length n=4m for three active users," *IEEE Transactions on Information Theory*, vol. 52, no. 8, pp. 2732–2742, 2007.

[4] V. I. Levenshtein, "Conflict-avoiding codes for three active users and cyclic triple systems," *Problems of Information Transmission*, vol. 43, no. 4, pp. 199–212, 2007.

[5] V. I. Levenshtein and V. D. Tonchev, "Optimal conflict-avoiding codes for three active users," *Proceedings of the IEEE International Symposium on Information Theory*, vol. 4-9, pp. 535–537, 2005.

[6] J. L. Massey and P. Mathys, "The collision channel without feedback," *IEEE Transactions on Information Theory*, vol. 31, no. 2, pp. 192–204, 1985.

[7] K. Momihara, "Necessary and sufficient conditions for tight equidifference conflict avoiding codes of weight three," *Designs, Codes and Cryptography*, vol. 45, no. 3, pp. 379–390, 2007.

[8] K. Momihara, M. Mishima, "A new series of optimal tight conflict-avoiding codes of weight 3," *Discrete Mathematics*, vol. 340, no. 4, pp. 617–629, 2017.

[9] J. Satoh, K. Momihara and M. Jimbo, "Constant weight conflict avoiding codes," *SIAM Journal on Discrete Mathematics*, vol. 21, no. 4, pp. 959–979, 2008.

[10] K. W. Shum and W. S. Wong, "A tight asymptotic bound on the size of constant-weight conflict-avoiding codes," *Designs, Codes and Cryptography*, vol. 57, no. 1, pp. 1–14, 2010.

[11] K. W. Shum, Y. Zhang and W.S. Wong, "Strongly conflict-avoiding codes," *SIAM Journal on Discrete Mathematics*, vol. 25, no. 25, pp. 1035–1053, 2011.

[12] B. S. Tsybakov and A. R. Rubinov, "Some constructions of conflict-avoiding codes," *Problems of Information Transmission*, vol. 38, no. 4, pp. 268–279, 2002.

[13] W. S. Wong, K. W. Shum, and C. S. Chen, "A general upper bound on the size of constant weight conflict avoiding codes," *IEEE Transactions on Information Theory*, vol. 56, no. 7, pp. 3265–3276, 2010.

[14] Z. T. C. Zhao, Z. Cheng, "The application of group in CAC constructions," *Journal of Anhui University (Natural Science Edition)*, vol. 43, no. 1, pp. 20–23, 2019.

[15] C. E. Zhao, W. P. Ma and D. S. Shen, "New optimal constructions of conflict avoiding codes of odd length and weight 3," *Designs, Codes and Cryptography*, vol. 73, no. 3, pp. 791–804, 2014.

Biography

Chun-e Zhao was born in Shandong Province, China, in 1981. She received the Ph.D. degree at Xidian university in 2015. She is now a lecture of China University of Petroleum. Her research interests include cryptogrophy coding and sequence design. (Email: zhaochune1981@163.com).

Yuhua Sun was born in Shandong province, China, in 1979. She received the Ph.D. degree at Xidian University in 2013. She is now a lecture of China University of Petroleum. Her research interests include cryptogrophy coding and sequence design.

Research on Secure Storage of Network Data Based on Cloud Computing Technology

Jinfeng Zhu

(Corresponding author: Jinfeng Zhu)

School of Computer Science and Engineering, Sanjiang University
No. 310, Longxi Road, Tiexin Bridge, Yuhuatai District, Nanjing, Jiangsu 210012, China
Email: ji18094@163.com

(Received July 10, 2020; Revised and Accepted Aug. 10, 2021; First Online Nov. 9, 2021)

Abstract

Cloud storage technology brought by cloud computing relieves the storage performance pressure of independent computers, but it also puts forward requirements for data storage security. This paper gave brief introductions to the cloud computing-based cloud storage, asymmetric encryption algorithm, the Advanced Encryption Standard (AES) algorithm, and an asymmetric encryption algorithm. This SM2 algorithm was used to ensure the security of cloud storage data. A hybrid encryption technology combining the AES algorithm and the SM2 algorithm was proposed. Finally, we tested the performance of the three encryption algorithms in cloud storage in the laboratory's local area network (LAN). The results showed that the AES algorithm spent the shortest time encrypting and decrypting data. The SM2 algorithm spent the longest time, and the time consumed by the AES+SM2 algorithm was significantly shorter than that of the SM2 algorithm and slightly longer than the AES algorithm. Considering that the complexity of the ciphertext decoded by the AES algorithm in the encryption performance experiment was the highest, followed by the SM2 algorithm and the AES+SM2 algorithm, the defect that the time consumed by the hybrid encryption technology was slightly longer than that by the AES algorithm in encryption and decryption was almost ignored. The AES algorithm failed in recognizing the tampering of ciphertext data in the cloud storage process, but the SM2 algorithm and the AES+SM2 algorithm succeeded.

Keywords: Asymmetric Encryption; Cloud Computing; Data Storage; Symmetric Encryption

1 Introduction

Due to the convenience of the Internet, people use the Internet more frequently for shopping, watching videos, and even banking [10]. Facing the increasing big data on the Internet, a single computer device has been difficult to meet the corresponding computing needs [3]. Cloud computing brought by high-speed and low latency communication technology provides an effective solution for the efficient processing of Internet big data [12]. When users use cloud computing to process big data, they no longer rely on the computing resources of their devices but upload the processing tasks to the cloud and use the distributed computing resources in the cloud to solve the tasks uploaded by users [9]. Cloud storage is an expanding service of cloud computing, which also uses distributed storage resources to store data in the cloud. Similarly, due to the good scalability, the cloud storage space is large enough, and the performance requirements for client equipment are low.

Cloud computing and cloud storage can provide users with computing and storage resources that are not limited by the performance of devices; however, important privacy information will also be transmitted in the process of use. Also, the open sharing characteristic of cloud services makes the stored data more vulnerable to intrusion and theft. Yoon *et al.* [14] proposed a density-based data encryption scheme and a database outsourcing query processing algorithm. In the performance analysis, compared with the existing schemes, the proposed scheme had better query processing performance and ensured the user's privacy. Xue and Ren [13] proposed an improved, efficient data encryption method, which was based on the ciphertext policy attribute-based encryption and used the fixed-length ciphertext to control the time cost. The simulation results showed that the improved algorithm had high reliability. Due to the traditional cloud computing-based information transmission mechanism and the problems of large errors and low security existing in and the Internet of things, Ding et al. [2] proposed a network resource management algorithm and verified the effectiveness of the algorithm through experiments.

2 Cloud Computing and Cloud Storage

Cloud computing [4] is a distributed data computing mode, which uses the data sharing function of the Internet to process a large number of data in a distributed way. These processes can be carried out in parallel; thus, it has higher computing efficiency. As cloud computing technology is based on the data sharing function on the Internet, although it makes the data processing more efficient, the openness brought by data sharing makes the data more vulnerable to attack. If they are public data, the loss will not be great; but if they are private data very important for users, the loss will be very serious.

Cloud storage is based on cloud computing technology, which not only improves the storage capacity but also inherits the characteristics of data sharing and data security protection in cloud computing [11]. The basic architecture of the cloud computing-based network storage system is shown in Figure 1. Users use the client to register or log in directly in the login interface and connect with the control center in the cloud server through the corresponding user interface to carry out corresponding system operations, including information registration, user login, data access, etc. After receiving the system operation information from the user interface, the control center obtains the corresponding service data from the node cluster, and the data of the node cluster come from the data pool composed of node clusters. Data pool is cloud computing space, usually composed of node clusters, and it can store data.

3 Storage Data Encryption

The growing big data contains a large number of valuable laws. The mining and analysis of the laws can provide effective guidance for activities such as production and operation. But these big data also contain a lot of privacy information, which would cause serious losses if they were used by lawbreakers. Therefore, in the cloud storage process, in addition to the need for identity authentication in the transmission process, storing data after encryption is also needed.

There are many kinds of data encryption algorithms in the cloud storage process, but the steps and models used in information transmission and storage encryption are nearly the same. Figure 2 is the basic model of transmission information encryption technology. First, at the sending node, the plaintext information to be sent is encrypted by the key generated by the encryption algorithm to obtain the ciphertext; then, the ciphertext is transmitted to the receiving node for storage through the secure channel of the Internet, and the key is transmitted to the receiving node through the non-secure channel [5]; the receiving node receives the ciphertext and stores it, and the ciphertext needs to be decoded using the key to obtain the plaintext before reading. In the basic model, the key and ciphertext are transmitted through different channels to reduce the probability that they are stolen at the same time. Ciphertext, the main body of data transmission, is further protected through the secure channel. As the key of encryption and decryption, the key itself does not have a specific meaning. As long as the key is not intercepted at the same time as the ciphertext, the plaintext will not be disclosed; therefore, the key can be transmitted by the non-secure channel to save the resource of the secure channel.

3.1 Advanced Encryption Standard Algorithm

It is seen from the previous description that cloud storage is based on cloud computing technology and uploads the data to be stored to the cloud composed of server node clusters. In order to prevent data leakage due to the data sharing characteristic in cloud storage and strengthen the security of data storage, the data are encrypted by an encryption algorithm and stored in the cloud [7].

Advanced encryption standard (AES) algorithm [1] is a kind of symmetric encryption technology. Considering that a 128-bit key is enough, it is usually used as the key of the AES algorithm. The encryption process of the AES algorithm is as follows.

1) The plaintext that needs to be encrypted is grouped, and the length of each group is 128 bytes. When it is less than 128 bytes, the group is supplemented.
2) The plaintext (128 bytes/group) and the key with the same number of bytes are put into two 4×4 matrices.
3) The plaintext matrix is encrypted by the key matrix.
4) The encrypted ciphertext matrix is processed by byte replacement, row displacement, and column mixing.
5) The key matrix is extended by using the key schedule function, and then the ciphertext matrix is encrypted again by using the extended key matrix.
6) Steps 4 and 5 are repeated ten times. Column mixing operation is not carried out in the tenth repetition. Finally, the ciphertext is obtained.

The decryption process of the AES algorithm is the reverse operation of the encryption process. In addition to the reverse operation of steps, the encryption of the information matrix is also reverse, including reverse byte replacement, reverse row displacement, reverse column mixing, etc.

3.2 SM2 Algorithm

SM2 algorithm [6] is a kind of asymmetric encryption technology, which uses a basic elliptic curve to generate key pairs. The encryption process is as follows.

1) Integer k generates randomly in $[1, n-1]$, where n refers to the order of base point G.

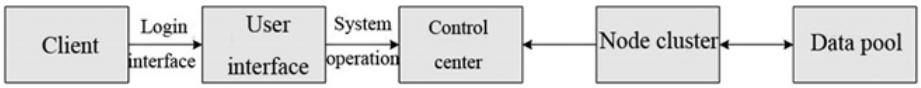


Figure 1: The basic architecture of the cloud computing-based network storage system

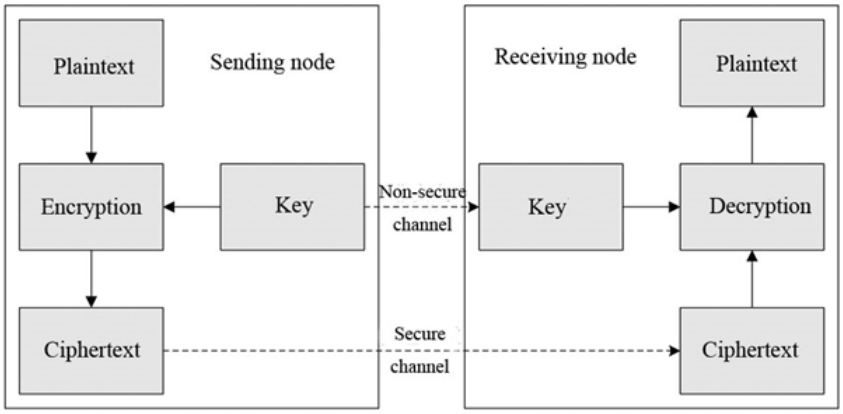


Figure 2: The basic model of information encryption technology in cloud storage

2) Points C_1, S, and Q in the elliptic curve are calculated according to Equation (1), and the formula is:

$$\begin{cases} C_1 = [k]G = (x_1, y_1) \\ S = [h]P_B \\ Q = [k]P_B = (x_2, y_2) \end{cases} \tag{1}$$

where h is a cofactor; (x_1, y_1) and (x_2, y_2) are coordinates of C_1 and Q, which need to be converted into bit string in the following steps, and P_B is a public key.

3) Bit string t of the key data is calculated:

$$t = DF(x_2||y_2, klen). \tag{2}$$

The decryption process is as follows.

1) Whether the equation of the elliptic curve is true or not is verified according to the coordinate point C_1 converted by bit string C_1 separated from ciphertext C. If it is true, decryption will continue; if it is not true, it means that the ciphertext is not encrypted by the SM2 algorithm, and encryption is no longer carried out.

2) Coordinate point C_1 is substituted into:

$$S = [h]C_1. \tag{3}$$

Whether point S is an infinite point in the elliptic curve is tested. If it is an infinite point in the elliptic curve, decryption will stop; if not, decryption will continue.

3) Key bit string t' used for decryption is obtained based on coordinate point C_1 and private key d_B. The calculation formula is:

$$\begin{cases} (x_2, y_2) = d_B C_1 \\ t' = KDF(x_2||y_2, klen). \end{cases} \tag{4}$$

4) C_2 in ciphertext C is decoded using t' to obtain plain text M'. The hash value of M' is calculated using the hash function of the SM3 algorithm. If the hash value of M' is the same with C_3, it indicates the information has not been tampered, and the decryption is successful. If the hash value is not the same with C_3, it indicates the information has been tampered, and the decryption fails.

3.3 Hybrid Encryption Algorithm

Although the AES algorithm has high efficiency in encryption and decryption, which is suitable for the big data environment under cloud computing, the single key is easier to be intercepted during transmission, and the receiver can not confirm whether the transmission data has been tampered with [8]. Only the public key will participate in the transmission of the SM2 algorithm; thus, even if the public key is intercepted, the third party can not decrypt by virtue of the public key, and the integrity of information can be verified by calculating the hash value of the information by the hash function, which greatly improves the security of transmission and storage. However, as asymmetric encryption technology, the SM2 algorithm is not as efficient as the AES algorithm in key generation, encryption, and decryption; thus, it is not suitable for the big data environment of cloud computing.

In order to improve the efficiency of encryption as much as possible on the premise of ensuring the storage security of transmission data, this paper uses hybrid encryption technology to encrypt the network data in the cloud computing environment.

The flow of hybrid encryption technology used in the cloud computing-based data storage process is shown in Figure 3, and the specific steps are as follows.

1) After the user enters the plaintext information in the client, the client will use a hash function, the SM3 algorithm, to calculate the hash value Z_1 of the plaintext and merge it with the plaintext.

2) The client uses the key of the AES algorithm to encrypt the merged information to obtain ciphertext C_1. The key of the AES algorithm was processed by SM2 encryption using the public key sent by the cloud server to obtain ciphertext C_2.

3) C_1 and C_2 are merged and uploaded to the cloud server.

4) After receiving the merged information, the cloud server splits it into C_1 and C_2.

5) The server uses the cloud private key pair to decrypt C_2 to obtain the key of the AES algorithm. C_1 is decoded by the key to obtain the ciphertext and the merged information of Z_1.

6) The hash value Z_2 is obtained by performing a hash calculation on the ciphertext using the SM3 algorithm. Z_1 is compared with Z_2. If they are not consistent, it indicates that the transmission data have problems, and the cloud server will not store the data; if they are consistent, the server will store the ciphertext.

In the above process, Steps 1, 2, and 3 are carried out in the user client, and Steps 4, 5, and 6 are carried out in the cloud server. Also, the decryption operation of the regular cloud storage service before storing the ciphertext in the cloud mainly aims to obtain the hash value of the decoded plaintext to verify whether there is a problem with the ciphertext information. The plaintext obtained in the verification process will not be stored in the server, and the ciphertext will be stored only after the verification is passed. Theoretically, there will be no plaintext in the cloud server.

4 Experimental Analysis

4.1 Experimental Environment

The experiment was carried out in the LAN of the laboratory, in which servers A and B were used as the storage nodes of cloud service, server C was used as the server for key generation, encryption, and decryption of cloud storage service, and a personal computer was used as the client. The server as the storage node had a 2 TB capacity. The server for key generation, encryption, and decryption of the cloud storage service had 16 G memory, a Core I7 processor, and a 64-bit Windows operating system. The personal computer was configured with 6 GB memory, a 64-bit Windows operating system, and a Core i7 processor.

4.2 Experimental Methods

1) One hundred packets in sizes of 5 MB, 15 MB, 25 MB, 35 MB, and 45 MB were set and uploaded to the cloud server using the personal computer client. In the uploading process, the data were encrypted by AES, SM2, and AES + SM2 algorithms. The time consumed by the three encryption algorithms in encryption and decryption was detected.

2) Another one hundred packets in sizes of 5 MB, 15 MB, 25 MB, 35 MB, and 45 MB were also set and uploaded to the cloud server by the personal computer client. Before the ciphertext was stored in the cloud server, the ciphertext was decoded by another server to simulate the situation that the information was cracked by a third party. The maximum cracking time was set as 60 min. The cracked ciphertext was compared with the original text to obtain the decryption integrity.

3) One hundred packets in sizes the same as in the previous two experiments were set. Before uploading to the cloud server through the client, the packets were uploaded to the third-party server. Without the corresponding key, the ciphertext decryption under the three encryption algorithms was difficult and could not be cracked in a short time; moreover, the cloud server would not store the plaintext after decoding the ciphertext. Therefore, the ciphertext was tampered with by adding 5-bit ciphertext in the original ciphertext packet and uploaded to the cloud server for storage to verify whether the three encryption algorithms could identify the tampered ciphertext data.

4.3 Experimental Results

In the cloud computing-based cloud storage service, the encryption of information is very important. In addition to ensuring the security of encryption, the efficiency of encryption and decryption is also an important factor. Figure 4 shows the encryption and decryption time of three encryption technologies in cloud computing. First of all, it was seen that the time consumed by the three encryption technologies increased with the increase of the data packet size, i.e., the increase of the amount of data to be processed. The decryption and encryption time of different algorithms differed little. The AES algorithm only used a single key, which was more efficient in key generation, encryption, and decryption. The key pair

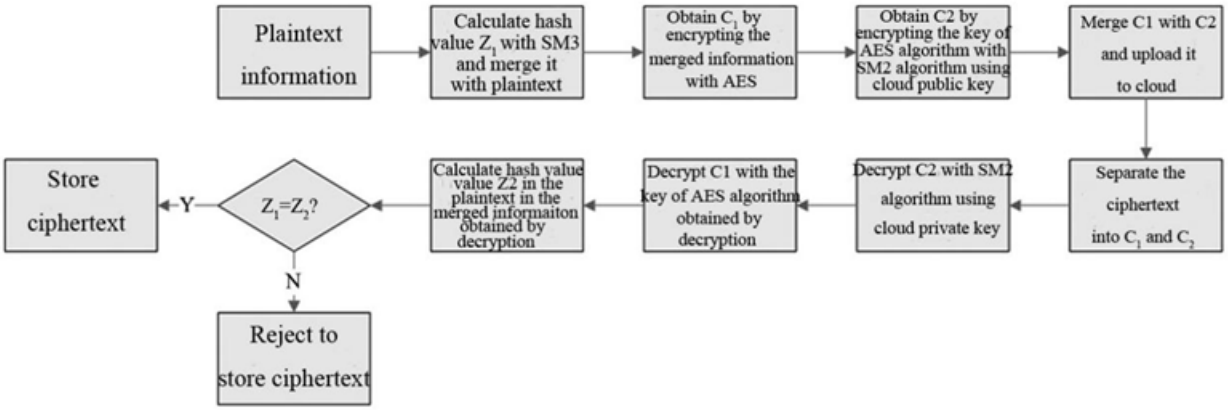


Figure 3: The flow chart of hybrid encryption technology

generation of the SM2 algorithm was more complex and time-consuming, and its encryption process and decryption process were not as reversible as the AES algorithm; thus, it was inefficient in the face of big data. The AES + SM2 hybrid algorithm used the SM2 algorithm for re-encryption based on the AES algorithm; thus, the time consumed was slightly longer than the AES algorithm; however, the SM2 algorithm was only used for encrypting the key of the AES algorithm, and the length of the key was significantly smaller than the plaintext. Thus, the time consumed by the hybrid algorithm was shorter than the single SM2 algorithm.

In the cloud computing-based cloud storage service, the confidentiality of encrypted information is very important. Figure 5 shows the decryption integrity of the data packet 60 minutes after being cracked by a third party under three encryption technologies. It was seen from Figure 4 that the decryption integrity of data packets under the three encryption technologies decreased with the increase of the data packet size. The increase of the data packet size meant that the amount of data increased, and the amount of calculation also increased during cracking; thus, the decryption integrity decreased under the limited decryption time. Comparing the data integrity of packets of the same size being processed by different encryption technologies, the AES algorithm was the highest, followed by the SM2 algorithm and the AES + SM2 algorithm. The reason for the above result is as follows. The AES algorithm only used single key encryption; thus, the encrypted ciphertext presented a relatively fixed law. The SM2 algorithm used public-key encryption; thus, the regularity of encrypted ciphertext was greatly reduced. The AES + SM2 algorithm not only encrypted the plaintext but also added the ciphertext to the key of the AES algorithm after asymmetric encryption, reducing the regularity of the ciphertext and improving the decryption difficulty.

The ciphertext packets that needed to be uploaded to the cloud server for storage were tampered with by a third-party server, and the detection performance of the three encryption technologies on the tampered data was tested. The results are shown in Table 1. It was seen from Table 1 that when the AES algorithm was used as the encryption technology for cloud storage, the tampered ciphertext packets of any size were successfully stored in the cloud server; the cloud storage service using the SM2 technology and the AES + SM2 encryption technology effectively detected and rejected the tampered ciphertext packets. The reason for the above results is as follows. The AES algorithm used single key encryption and decryption without an additional verification mechanism. Even if the encrypted ciphertext was decrypted to plaintext, if there was no actual plaintext for comparison, it was impossible to confirm whether the data has been tampered with. If the plaintext was sent for verification, then encryption technology was meaningless. The SM2 algorithm could verify by taking the hash value of the plaintext as a comparison basis. The AES+SM2 algorithm combined the two algorithms, thereby inheriting the data integrity detection mechanism in the SM2 algorithm. In that mechanism, it did not need to send plaintext as a comparison but used hash values representing the complete plaintext to ensure the security of plaintext.

5 Conclusion

This paper gave brief introductions to the cloud computing-based cloud storage, a symmetric encryption algorithm, the AES algorithm, and an asymmetric encryption algorithm, the SM2 algorithm. Combining the two algorithms, the AES + SM2 hybrid encryption technology was proposed. Finally, the performance of the three encryption algorithms in cloud storage was tested in the LAN of the laboratory. The results are as follows.

1) With the increase of the cloud storage packet size, the encryption and decryption time of the three encryption algorithms decreased; when processing the packet of the same size, the AES algorithm had the shortest encryption and decryption time, the

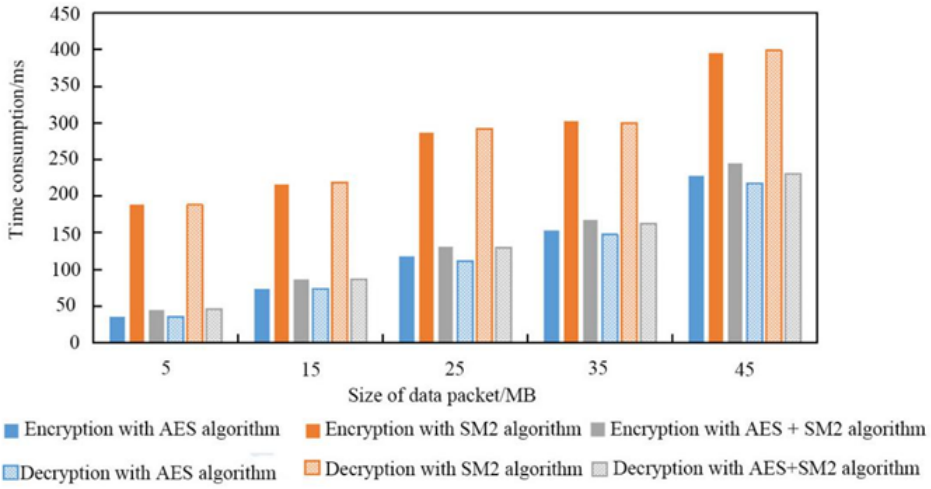


Figure 4: Time consumed by three encryption technologies in encryption and decryption in cloud computing

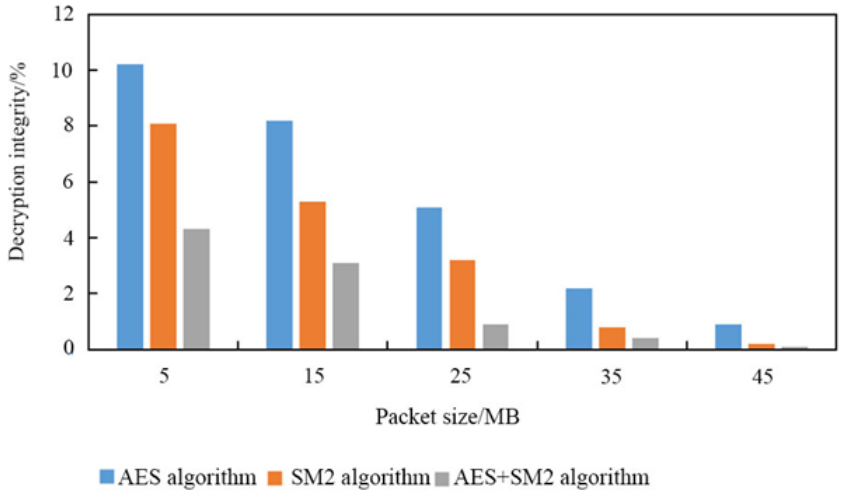


Figure 5: Security of three encryption technologies in cloud computing

Table 1: Detection performance of three encryption technologies for tampered ciphertext

	5 MB	15 MB	25 MB	35 MB	45 MB
AES	Storage succeeds	Storage succeeds	Storage succeeds	Storage succeeds	Storage succeeds
SM2	Storage fails	Storage fails	Storage fails	Storage fails	Storage fails
AES+SM2	Storage fails	Storage fails	Storage fails	Storage fails	Storage fails

SM2algorithm had the longest encryption and decryption time, and the time of the AES + SM2 algorithm was between the two other two algorithms.

2) With the increase of the data packet size, the integrity of data under the three encryption technologies decreased, and when processing the data packet of same size, the integrity of data under the AES algorithm was the highest, followed by the SM2 algorithm and the AES + SM2 algorithm.

3) In the face of tampered data, no matter how big the data packet was, the cloud storage service under the AES encryption technology could not make effective identification, but the cloud storage service under SM2 and AES + SM2 encryption technologies could make effective identification.

References

[1] A. Banushri, R. A. Karthika, "A Survey on data security using file hierarchy attribute-based encryption in cloud computing environment," *Journal of Advanced Research in Dynamical & Control Systems*, vol. 2017, no. 4, pp. 144-149, 2017.

[2] L. Ding, Z. Wang, X. Wang, D. Wu, "Security information transmission algorithms for IoT based on cloud computing," *Computer Communications*, vol. 155, pp. 32-39, 2020.

[3] G. M. Kiran, N. Nalini, "Enhanced security-aware technique and ontology data access control in cloud computing," *International Journal of Communication Systems*, vol. 33, no. 15, pp. e4554, 2020.

[4] K. S. Kumar, K. Shankar, M. Ilayaraja, M. Rajesh, "Sensitive data security in cloud computing aid of different encryption techniques," *Journal of Advanced Research in Dynamical and Control Systems*, vol. 9, no. 18, pp. 2888-2899, 2018.

[5] R. Manoharan, "Sensitive data security in cloud computing aid of different encryption techniques," *Journal of Advanced Research in Dynamical and Control Systems*, vol. 18, pp. 2888-2899, 2017.

[6] P. R. More, S. Y. Gaikwad, "An advanced mechanism for secure data sharing in cloud computing using revocable storage identity based encryption," *International Journal of Engineering Business Management*, vol. 1, no. 1, pp. 12-14, 2017.

[7] V. Naresh, M. Anudeep, M. Saipraneeth, A. S. Reddy, V. Navya, "Encryption-based secure and efficient access control to out sourced data in cloud computing," *International Journal of Engineering & Technology*, vol. 7, no. 2, pp. 315, 2018.

[8] T. Paka, S. Divya, "Data storage security and privacy in mobile cloud computing using hierarchical attribute based encryption (HABE)," *International Journal of Computer Sciences and Engineering*, vol. 7, no. 6, pp. 750-754, 2019.

[9] K. Rangasami, S. Vagdevi, "Comparative study of homomorphic encryption methods for secured data operations in cloud computing," in *International Conference on Electrical*, pp. 1-6, 2017.

[10] P. Sawle, T. Baraskar, "Survey on data classification and data encryption techniques used in cloud computing," *International Journal of Computer Applications*, vol. 135, no. 12, pp. 35-40, 2016.

[11] M T. Sultan, K. N. Yasen, "Homomorphic encryption implementation to ensure data security in cloud computing," *Journal of Theoretical and Applied Information Technology*, vol. 96, no. 7, pp. 1826-1836, 2018.

[12] L. Teng, H. Li, S. Yin, Y. Sun, "A modified advanced encryption standard for data security," *International Journal of Network Security*, vol. 22, no. 1, pp. 112-117, 2020.

[13] S. Xue, C. Ren, "Security protection of system sharing data with Improved CP-ABE encryption algorithm under cloud computing environment," *Automatic Control and Computer Sciences*, vol. 53, no. 4, pp. 342–350, 2019.

[14] M. Yoon, M. Jang, Y. S. Shin, J. W. Chang, "A bitmap based data encryption dcheme in cloud computing," *International Journal of Software Engineering & Its Applications*, vol. 9, no. 5, pp. 345-360, 2015.

Biography

Jinfeng Zhu,born in 1980, holds a bachelor's degree from Southeast University. Now he works in Nanjing Sanjiang University and is the director of the professional laboratory. Mainly engaged in the construction and management of professional laboratories, and participated in the teaching and scientific research of computer science and technology, software engineering, network engineering and other majors. Research interests: computer application, multimedia technology, cloud computing, etc.

Frequent Itemset Mining Algorithm Based on Differential Privacy in Vertical Structure

Shigong Long, Hongqin Lu, Tingting Chen, Nannan Zhou, and Hai Liu
(Corresponding author: Tingting Chen)

Department of Computer Science and Technology, Guizhou University
Guizhou provincial Key Laboratory of Public Big Data, Guizhou University
Gui Yang 550025, China
Email: 769176017@qq.com

(Received Dec. 8, 2020; Revised and Accepted May 26, 2021; First Online Dec. 22, 2021)

Abstract

Frequent itemset mining is an important aspect of data mining. Direct publication of frequent item-set mining results may cause serious personal privacy leakage, so this article proposes a frequent item-set mining algorithm that satisfies differential privacy. This article proposes a weight truncation algorithm for transaction truncation to reduce the global sensitivity of differential privacy. The truncated transaction retains the frequent item information of the original transaction as much as possible. This article uses average support and maximum support estimation strategies to reduce errors due to transaction truncation and pruning operations. The experimental results are compared and analyzed, and the proposed algorithm satisfies the differential privacy protection. At the same time, the frequent itemset mined has good utility. The proposed algorithm has higher availability than the existing algorithm TT (Transaction Truncation) and algorithm PB (PrivBasis).

Keywords: Differential Privacy; Frequent Itemset; Support Estimation Strategies; Transaction Truncation

1 Introduction

With the continuous popularization and application of information technology, data mining brings convenience to people's lives, but also brings privacy risks. Frequent itemset mining is an important part of data mining. Agrawal proposed Apriori algorithm [1] is the most frequent itemset mining algorithm, but the algorithm needs to repeatedly scan the database, candidate set generation and testing efficiency is low, and takes up a lot of memory. The FP-growth algorithm proposed [8] by Han *et al.* does not generate a candidate set. By compressing the data set into a frequent pattern tree, the frequent item set is generated using the pattern growth method. Therefore, the efficiency is improved compared to the Apriori algorithm, but recursive Generating conditional databases and conditional FP-trees takes more time and memory.The above algorithm uses horizontal data representation to represent the database. Zaki's Eclat algorithm [16] uses vertical data representation to represent the database. The performance of algorithms using vertical data representation is generally superior to that of horizontal data representation, because it only needs to search the database once.The local differential privacy algorithm used by Wang *et al.* [14]. Ensures that there is no trusted third party, but does not use truncation algorithm, which makes the local processing inefficient.

Differential privacy [4] is a new approach to privacy protection proposed by Dwork in 2006 for the privacy database leakage problem. Differential privacy can solve two shortcomings of the traditional privacy protection model. First of all, the differential privacy protection model assumes that the attacker can obtain all other recorded information except the target record. The sum of these information can be understood as the maximum background knowledge that the attacker can grasp. Under this maximum background knowledge assumption, differential privacy protection does not need to consider any possible background knowledge possessed by the attacker, because these background knowledge cannot provide more abundant information than the maximum background knowledge. Second, it is based on a solid foundation of mathematics. It strictly defines privacy protection and provides quantitative assessment methods, making the privacy protection level provided by data sets processed by different parameters comparable.

2 Preliminaries

2.1 Differential Privacy

Differential privacy provides a strong privacy guara ntee that the output of a computation to neighbori ng databases if they differ by at most one record. Formally, the differential privacy is defined as follows.

Definition 1. *(ϵ-differential privacy [4]). A private algorithm P satisfies ϵ-differential privacy for any two neighboring databases D and D′, and any subset of outputs S⊆Range(P), $Pr[P(D) \subseteq S] \leqslant e^{\epsilon} Pr[P(D^{'}) \subseteq S]$, Where the probability is taken over the randomness of P.*

For any pair of neighboring databases, differential privacy guarantees that the difference in the probability of any subset of outputs P is bounded by e^{ϵ}, where smaller values of ϵ mean a stronger privacy guarantee.

Definition 2. *(Sensitivity [6]). Given n count queries Q, for any neighboring databases D, D′, the sensitivity of Q is* $\Delta Q = max \left\| Q(D) - Q(D^{'}) \right\|_1$.

Theorem 1. *(Laplace mechanism). Let Q denote a query sequence of length p with sensitivity ΔQ. Let ψ_1,...,ψ_p be a p-length vector, where ψ_i is drawn i.i.d. from a Laplace distribution with scale $\frac{\Delta Q}{\epsilon}$. Then, the algorithm: $P(D) = Q(D) + \{\psi_1, ..., \psi_p\}$ Achieves ϵ-differential privacy.*

The Laplace distribution with magnitude λ, *i.e.*, Lap(λ), follows the probability density function as $Pr[x|\lambda] = \left(\frac{1}{2\lambda}\right) e^{\frac{-|x|}{\lambda}}$, where $\lambda = \frac{\Delta Q}{\epsilon}$ is determined by both the sensitivity ΔQ and the privacy budget ϵ.

Theorem 2. *(Geometric mechanism [5]). Let Q be a query sequence of length p with integer outputs, and its sensitivity is ΔQ. The algorithm: $P(D) = Q(D) + \{\psi_1, ..., \psi_p\}$ gives ϵ-differential privacy, where ψ_i i.i.d. samples from a distribution $G\left(\frac{\epsilon}{\Delta Q}\right)$.*

The magnitude of injected noise conforms to a two-sided geometric distribution G(α) with the probability mass function: $Pr[x|\alpha] = ((exp(\alpha) - 1) / (exp(\alpha) + 1)) exp(\alpha)^{-|x|}$ where $\alpha > 0$.

Theorem 3. *(Sequential Composition). Let f_1,$\cdots$,f_m be m randomized algorithms, where f_i provides ϵ_i-differential privacy ($1 \leq i \leq m$). A sequence of $f_i(D)$ over database D provides($\sum \epsilon_i$)-differential privacy.*

2.2 Frequent Itemset Mining

Given the alphabet $\xi = \{i_1, \cdots, i_n\}$, a transaction t is a subset of ξ and a transaction databases D is a multiset of transactions [15]. Each transaction represents an individual′s record. A non-empty set $X \subset \xi$ is called an itemset. The length of an itemset is the number of items in it. An itemset is called a k-itemset is if it contains k items. We say a transaction t contains an itemset X if X is a subset of t. The support of itemset X is the number of transactions containing X in the databases. An itemset is frequent if its support is no less than the user-specified minimum support threshold. Given a transaction database and a user-specified minimum support threshold, the goal of FIM is to find the complete set of frequent itemsets. In the rest of this paper, we use "threshold" as shorthand for "user-specified minimum support threshold".

2.3 Vertical Mining Algorithm

Many vertical mining algorithm have been proposed recently for association mining, which have shown to be very effective and usually outperform horizontal [9] approaches. The main advantage of the vertical format [7] is support for fast frequency counting via intersection operations on transaction tids and automatic pruning of irrelevant data.

Next we introduce the vertical structure algorithm used in this paper. The original dataset is shown in Table 1. Firstly, statistics are made for 1-itemset, assume the threshold is 0.5. And 1-itemsets of vertical structure tables is created for frequent items that are larger than the threshold. As shown in Table 2. Connect according to 1-itemset vertical structure tables, and perform transaction intersection operations on transaction numbers, retain 2-itemset that are greater than the threshold, and build 2-itemsets tables. As shown in Table 3. Connect the 2-itemset and take pruning to get candidate 3-itemsets. Performing transaction intersection operations on transaction numbers of the 2-itemsets that make up the candidate 3-itemsets, retaining 3-itemsets that are greater than the threshold, and building 3-itemsets tables, as shown in Table 4. And so on until no new itemsets are generated.

3 Algorithm Design

In this section, we present a straightforward approach to make vertical mining achieve ϵ-differential privacy. Our main idea is to add noise to the support of itemsets and use their noisy supports to determine which itemsets are frequent. Suppose the maximal length of frequent itemsets is L_f. We uniformly assign the support computations of i-itemsets a privacy budget $\frac{\epsilon}{L_f}$. Then we add Laplace noisy $L(\Delta Q_i / (\epsilon / L_f))$ to the support of items. If the noisy support of an item exceeds the threshold, we output it as a frequent itemset. The sensitivity of Q_i is no greater than $min\left\{\binom{m}{i}, n\right\}$, where m is the maximum length of the transaction and n is the total number of candidate sets. To limit the cardinality of transactions′ candidate, we propose a transaction truncating approach. If a transaction has more than a specified number of items, items are deleted until the transaction is under the limit.

3.1 Design Overview

To discover frequent k-itemsets, we utilize frequent (k-1)-itemsets to generate candidate k-itemsets based on the downward closure property. Intersection operation of the vertical structure of frequent k-1 itemsets that make up a candidate k-itemsets, and use the noisy support of each candidate k-itemsets to determine whether it is frequent. We truncate the long transaction according to the weight of 1-itemsets.

The first four lines of algorithm 2 show how to find frequent k-itemsets in DP-verticals. Given the original

Table 1: Original dataset table

TID	Itemsets
1	A,B,C,D,E
2	A,B,D,F
3	A,C,E
4	B,C,F
5	C,D,F
6	A,B,D

Table 2: 1-itemset table

item	TID_set
A	1,2,3,6
B	1,2,4,6
C	1,3,4,5
D	1,2,5,6
E	1,3,5

Table 3: 2-itemsets table

item	TID_set
A,B	1,2,6
A,D	1,2,6
B,D	1,2,6
C,E	1,3,5

Table 4: 3-itemsets table

item	TID_set
A,B,D	1,2,6

database and candidate k-itemsets, we first transform the database to enforce the length constraint. For the transaction whose cardinality violates the constraint, we truncate it by using our weighted truncating algorithm. For candidate 1-itemset c_1, We create a vertical table of candidate 1-itemsets by searching the transformed database. We compute the noise support of 1-itemsets through the vertical structure table. Then we get a vertical table of 1-itemsets. For each candidate k-itemsets c_k, we compute its noisy support through the vertical structure table of (k-1)-itemsets. Based on the noisy support of c_k, we estimate the actual support of c_k in the original database by using support estimation technique. If the estimated "maximal" support of c_k exceeds the threshold, we keep its vertical structure table to generate candidate (k+1)-itemsets; if the estimated "average" support of c_k exceeds the threshold. We regard it as a frequent k-itemsets.

3.2 Weighted Truncating and Support Estimation

If we limit the cardinality of transactions by transaction truncating, we can keep more frequency information. The noise support of candidate 1-itemsets are arranged in descending order, and the m itemsets with the largest noise support in the long transaction is formed into a short transaction. This is because the combination of 1-itemsets with higher support is more likely to be frequent itemsets.

Despite the potential advantages of truncating the transaction, it inevitably incurs information loss. because the support of some itemsets decreases. Suppose a transaction $t = \{a, b, c\}$ is truncated that transaction to be $\{a, b\}$, the support of the itemset $\{a, b\}$ and $\{a, b\}$ change from 1 to 0.

To quantify this information loss, inspired by the double standards method [17] and Run-time Estimation [11]. In order to adapt to the algorithm in this paper, we have changed double standards method and Run-time Estimation. The method consists of two steps:

1) We estimate the actual support based on noise support in the transformed database.

2) We estimate the actual support in the original database based on the solid support in the transformed database.

For each itemset, we estimate its "average" support and "maximum" support, "average" support determines whether it is a frequent item, The "maximum" support determines whether it is used to generate the next candidate itemsets.

In the first step. Let α denote its noisy support in the transformed database and β denote its actual support in the transformed database. Based on the Bayesian rule, We can get this formula $Pr(\beta|\alpha) = Pr(\alpha|\beta) Pr(\beta) / Pr(\alpha)$. We assume a uniform prior on $Pr(\beta)$, we can see the probability distribution of β follow [17]:

$$Pr(\beta|\alpha) \sim e^{-\epsilon|\beta-\alpha|} \tag{1}$$

In the second step. According to the actual support in the transformed database, we estimate the actual support in the original database.

For a transaction t of length p, after truncating, the probability of an i-itemset X remaining in the transaction is, According to the combination theorem, We can get the formula: $\gamma_p = C_{p-1}^{m-i} / C_p^m$.

We quantify the "average" support information loss caused by truncating transactions. Suppose the size of the alphabet is n. Let $\mu(X) = (\mu_1, \mu_2, \cdots, \mu_n)$, where μ_k is the number of transactions with length k containing i-itemset X. We refer vector $\mu(X)$ to as the μ-vector of X. We consider the support of i-itemset X in the transformed database as a random variable. Then, the expectation of β is:

$$\begin{aligned} E(\beta) &= \sum_{k=i}^{m} \omega \cdot \frac{\mu_k}{\sum_{j=i}^{n} \mu_j} + \sum_{k=m+1}^{n} \omega \cdot \frac{\mu_k}{\sum_{j=i}^{n} \mu_j} \cdot \gamma_p \\ &= \omega \cdot (\sum_{k=i}^{m} \frac{\mu_k}{\sum_{j=i}^{n} \mu_j} + \sum_{k=m+1}^{n} \frac{\mu_k}{\sum_{j=i}^{n} \mu_j} \cdot \gamma_p). \end{aligned}$$

Let $ratio(i) = \sum_{k=i}^{m} \frac{\mu_k}{\sum_{j=i}^{n} \mu_j} + \sum_{k=m+1}^{n} \frac{\mu_k}{\sum_{j=i}^{n} \mu_j} \cdot \gamma_p$. Thus, given the actual support of X in the transformed database, we can estimate the "average" support of X in the original database to be

$$\beta_{avg} = avg(\beta, i) = \frac{\beta}{ratio(i)} \quad (2)$$

According to double standards method [17], we could estimate "maximal" support of X in the original database to be

$$\max(\beta, i) = \begin{cases} \frac{\beta - \ln\rho + \sqrt{\ln^2\rho - 2\beta\ln\rho}}{\text{ratio(i)}} & \text{if } \ln\rho \le 2\beta \\ \text{avg}(\beta, i) & \text{if } \ln\rho > 2\beta \end{cases} \quad (3)$$

By associating the first steps and second steps,we can compute the average original support by:

$$\omega_{avg} = avg\_supp(\alpha, i) = \int_{\beta=-\infty}^{\beta=+\infty} Pr(\beta|\alpha) avg(\beta, i)$$

and the maximal original support by:

$$\omega_{max} = \max\_supp(\alpha, i) = \int_{\beta=-\infty}^{\beta=+\infty} Pr(\beta|\alpha) \max(\beta, i).$$

When we get the noisy support of an i-itemset X, we estimate its "average support" to determine whether X is frequent. And, by estimating the "maximal support" of X, we decide whether to use X to generate the candidates of (i+1)-itemsets.

4 Algorithm Description and Privacy Analysis

The algorithm consists of two phases [17]. In the preprocessing phase, we extract some statistical information from the original database and leverage the weight truncating method to transform the database. The preprocessing phase is only execut once. In the mining phase, for a given the specified threshold, we privately find frequent itemsets [13]. We divide the total privacy budget ε into five portions. In order to improve the accuracy and privacy of the algorithm, geometric noise is added in the preprocessing process and Laplace noise is added in the generation of frequent itemsets.

4.1 Preprocessing Phase

We first compute the maximal length constraint L_m enforced in the database. Suppose the size of the alphabet is n. Let $\alpha = \{\alpha_1, \cdots, \alpha_n\}$, where α_i is the number of transactions with length i. We set Lm to the value such that the percentage $\sum_{i=1}^{L_m} \alpha_i / \sum_{i=1}^{n} \alpha_i$ is least η. Because the computation of α has privacy risk, experiments show that adding laplace noise in the preprocessing stage will reduce the accuracy of the algorithm, we add geometric noise $G(\epsilon_1)$ to each α_i. We compute $\beta = \{\beta_1, \cdots, \beta_n\}$, where β_i is the maximal support of i-itemsets. We use array β to estimate the maximal length of frequent itemsets L_f. Set r as the maximal length of discover ed frequent itemsets. For i from 1 to r, we get the maximal support of i-itemsets β_i . Since β is a property of the database, we add geometric noise $G(\epsilon_2/logn)$ to each β_i [11].

Algorithm 1 Weight_truncating

Input:
Original database D;
Privacy budget ϵ_3;
cardinality L_m;
Output:
Transformed database D';

```
Begin
Set=compute the noisy support of all 1-itemsets in D using ε3;
for each transaction t ∈ D do
  if |t| > Lm then
    SubTransactions ST = m itemsets with the largest noise support in the long transaction;
    add ST into D';
  else
    Add transaction t into D';
  end if
end for
End
```

We generate an $r \times n$ matrix Z to compute β, where row z_i is the μ-vector of the i-itemset with the highest support. It will be used in support estimation method to quantify the information loss caused by truncating transaction [11].

After that, we transform the database by using our weight truncating method. We use 1-itemsets support to take truncated long transactions. Since there is a risk of privacy leakage in the calculation of 1-itemsets support, we add geometric noise $G(\epsilon_3)$ to each 1-itemset support.

4.2 Mining Phase

In the mining phase, set the threshold to λ, we estimate the maximal length of frequent itemsets L_f. We set L_f to be the integer l such that β_l is the mining process, we use the support estimation method to quantify the information loss caused by truncating transaction. Since Z is a property of the database, to avoid the risk of privacy disclosure, for i from 1 to L_f, we add geometric noise $G(\epsilon_4/L_f)$ to each element in rowz_i of Z [11].

In order to obtain noise support of items,we add laplace noise $Lap(\Delta Q/(\epsilon_5/L_m))$ to the support computations of i-itemsets.Based on its "average" and "maximal" supports in the original database. If the estimated "maximal support" exceeds threshold λ, we reserve X for generating candidate itemsets. If the estimated "average support" exceeds λ, we output item X as a frequent itemsets.

Algorithm 2 DP-vertical

Input:
Database D;
maximal cardinality m;
privacy parameter $\epsilon_1, \epsilon_2, \epsilon_3, \epsilon_4, \epsilon_5$;
threshold λ;

Output:
The set of frequent itemsets F;

Begin
α=get noisy number of transactions with different length using ϵ_1;
L_m=get maximal length constraint L_m based on α and η;
β=get noisy maximal support of itmsets of different lengths using ϵ_2;
Z=compute an $r \times n$ matrix using the μ-vectors of itemsets using ϵ_4;
L_f=estimate maximal length of frequent itemsets based on β and λ;
$D' = \text{weight\_truncate}(D, m, \epsilon_3)$;
$c_1(item, value)$=set vertical structure table of 1-itemsets through D';
for each candidate itemset $X \in c_1$ **do**
  n_sup = compute_noisy_support(X_value,c_1_value, ϵ_5/L_f);
  max_sup = estimate_maximal_support(n_sup);
  if $max\_sup \geq \lambda$ **then**
    $F_i += X$;
  end if
  avg_sup = estimate_average_support(n_sup);
  if $avg\_sup \geq \lambda$ **then**
    $F += X\_item$;
  end if
end for
while F_i not $NULL$ **do**
  i=i+1;
  c_i=Generate candidate i-itemsets based on F_{i-1}
  for each candidate itemset X in c_i **do**
    n_sup=compute_noisy_support(X_value,c_1_value, ϵ_5/L_f);
    max_sup = estimate_maximal_support(n_sup);
    if $max\_sup \geq \lambda$ **then**
      $F += X\_item$;
    end if
    avg_sup = estimate_average_support(n_sup);
    if $avg\_sup \geq \lambda$ **then**
      $F += X\_item$;
    end if
  end for
end while
End

4.3 Privacy Analysis

In the preprocessing phase, for the computation of α, as a single transaction only affects one element in α by one, this computation's sensitivity is 1. Thus, adding geometric noise $G(\epsilon_1)$ in computing α satisfies ϵ_1-differential privacy. The maximal length constraint L_m is calculated according to α, so ϵ_1-differential privacy is satisfied. Moreover, adding geometric noise $G(\epsilon_2/logn)$ in computing β satisfies ϵ_2-differential privacy. We need the noisy support of 1-itemsets as a truncation indicator [12]. In order to protect privacy, adding geometric noise $G(\epsilon_3)$ to the support of each 1-itemset satisfies ϵ_3-differential privacy. In the mining phase. The maximal length of frequent itemsets L_f is calculated according to β, so $\epsilon_2/logn$-differential privacy is satisfied. For the computation of row $z_i.$ in Z, as adding or removing one transaction can only affect a single element in $z_i.$, so, the sensitivity of this computation about Z is 1. Thus, adding geometric noise $G(\epsilon_4/L_f)$ in computing $z_i.$ guarantees ϵ_4-differential privacy. According to the sequential composition property [5], matrix Z satisfies ϵ_4-differential privacy [11]. After that, we privately find frequent itemsets based on the database transformed by our weighted truncating operation. For i from 1 to L_f, we uniformly assign the support computations of i-itemsets a private budget $\epsilon' = \epsilon_5/L_f$. The sensitivity used to perturb the support of i-itemsets is ΔQ_i. Thus, adding Laplace noise $Lap(\Delta Q_i/\epsilon')$ to the support of each item satisfies ϵ'-differential privacy. In summary, our algorithm satisfies $\left(\epsilon = \sum_{i=1}^{5} \epsilon_i\right)$-differential privacy.

5 Experiments

5.1 Experiments Setup

We compare our DP-vertical algorithm with the following two algorithms. The Apriori-based algorithm which privately find itemsets whose support exceeds a given threshold [11]. And the "*PrivBasis*" algorithm which privately find the k most frequent itemsets [10]. We use DPVM to denote out the algorithm, while TT and PB to denote the algorithms Transaction Truncation and PrivBasis, respectively.

We implement these three algorithm in JAVA and conduct all experiments on a PC with Intel Core i7-6700HQ CPU(2.60GHz) and 16GB RAM. Since the algorithms involve randomization, we run each algorithm 10 times and report its average result. We set parameter ρ to be 0.01. The privacy budget ϵ is set to be 1.0. The parameter η is set to be 0.85.

The experiment uses datasets Accident and Pumsb_star widely used in frequent project set mining algorithms [3]. Table 5 summarizes the parameter settings for the data sets used in this experiment.

To evaluate the performance of our algorithm, we employ the widely used standard metrics. We use F-score [9] to measure the utility of generated frequent itemsets. It is defined as

$$F - score = 2 \times \frac{precision \times recall}{precision + recall}$$

Where $precison = |U_p \cap U_c| / |U_p|$, $recall = |U_p \cap U_c| / |U_c|$, U_p is the frequent itemsets gener-

ated by a private algorithm and U_c is the actual frequent itemsets.

And, to measure the error with respect to the actual supports of itemsets, we calculate the relative error of released itemset supports [9]

$$RE = median_X \frac{|sup'_x - sup_x|}{sup_x}$$

Where X is the generated frequent itemsets, supx is the actual support of itemset x and $supx'$ is the noisy support of itemset x.

Table 5: original dataset table

Dataset	Transaction	Items	Max.len	Avg.len
Pumsb-star	49046	2088	63	50.5
Accident	340183	468	51	33.8

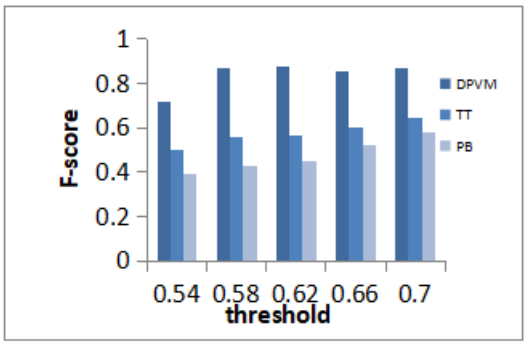


Figure 1: PUMSB: F-score with data set Pumsb-star

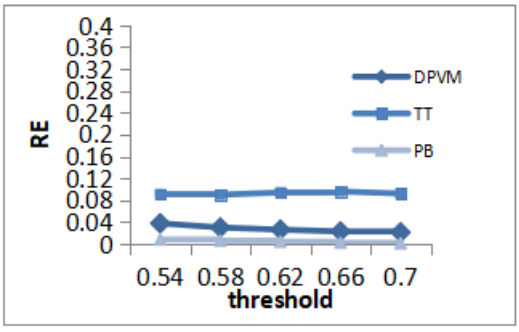


Figure 2: PUMSB: RE with data set Pumsb-star

5.2 Experiment Result

The privacy budget ϵ of Figures 1, 2, 3, and Figure 4 are set to 1, which shows the comparison of the F-score parameter and the RE parameter of the thresholds of the algorithms DPVM, TT, and PB from 0.54 to 0.7. Figure 1 and Figure 2 are comparisons of the data set Pumsb-star, and Figures 3 and 4 are comparisons of the data set Accident.

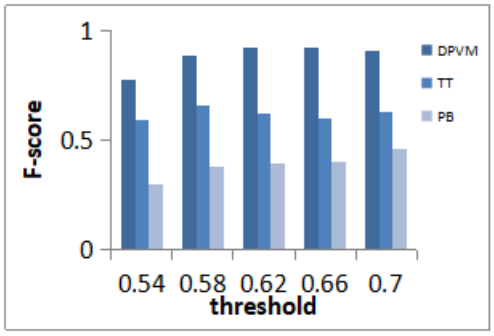


Figure 3: Accident: F-score with data set Accident

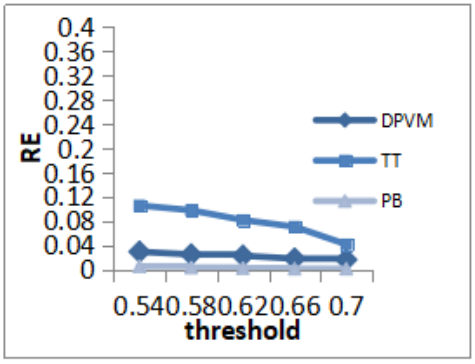


Figure 4: Accident: RE with data set Accident

It can be seen from the figure that the F-score parameter value of the algorithm DPVM is larger than the algorithms TT and PB under the same threshold, and the RE parameter value of the algorithm DPVM is smaller than the algorithm TT.

Figures 5 and 6 set the threshold to 0.6, which shows the comparison of the F-score parameter and the RE parameter of the privacy budget ϵ of the algorithm DPVM, TT, PB from 0.5 to 1.25.

It can be seen from the figure that under the same privacy budget ϵ, the F-score parameter value of the algorithm DPVM is larger than the algorithm TT, PB, and the RE parameter value of the algorithm DPVM is smaller than the algorithm TT.

As shown from the figure, the DPVM achieves better performance than TT on pumsb star and accidents. This is because DPVM retains more frequent information than TT by using weight truncation algorithm and support estimation strategy. We can also observe that DPVM is better than PB in parameter F-score on pumsb star and account datasets. Privbasis method uses the fact that all subsets of a frequent itemset are frequent to achieve dimension reduction. To find the most frequent K itemsets, the privbasis method finds the itemset IB, decomposes IB into several subsets, and proposes the concept of θ - basis set. When the support difference between item sets is very small, It is likely that infrequent items will be sampled, resulting in low F-score.

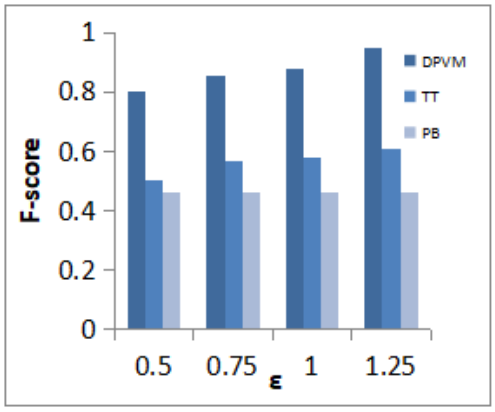


Figure 5: PUMSB: F-score with 0.6 threshold

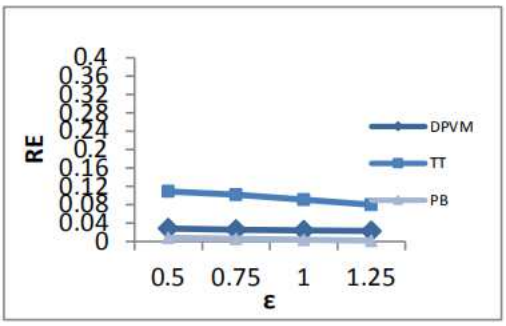


Figure 6: PUMSB: RE with 0.6 threshold

6 Conclusions

This paper studies the privacy leakage problem of frequent itemsets mining algorithm based on a vertical structure.it proposes a frequent itemsets mining algorithm DPVM based on differential privacy. By adding noise to the frequency of itemsets, the support of published itemsets is guaranteed to meet the differential privacy. In the case of ensuring that the privacy of the original data will not be disclosed, the amount of added noise is reduced. The data availability is improved. In this algorithm, the weight truncation algorithm is proposed to solve the large noise caused by long transactions. The support estimation strategy is adopted to reduce the error caused by truncation [2]. By reducing the amount of added noise, the accuracy of the algorithm is improved. Finally, the DPVM algorithm is compared with TT and PB algorithm, and a large number of experiments show that the algorithm has high availability under the condition of differential privacy.

In this paper, the differential privacy protection mechanism is studied, and differential privacy is applied to frequent itemset mining algorithms to protect the algorithm's privacy and achieve better usability. However, adding a differential privacy protection mechanism will cost more time and reduce the efficiency of mining large data sets. Therefore, how to ensure the availability of the differential privacy algorithm, reduce the time consumed by the algorithm, and improve the algorithm's efficiency is the next problem to be studied.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (NO.62062020,NO.62002081), the Major Scientific and Technological Special Project of Guizhou Province (Grant NO.20 183001).The authors gratefully acknowledge the anonymous reviewers for their valuable comments.

References

[1] R. Agrawal, T. Imieliński, and A. Swami, "Mining association rules between sets of items in large databases," in *Proceedings of ACM SIGMOD International Conference on Management of Data*, pp. 207–216, 1993.

[2] H. L. Chen, Y. T. Tsou, B. C. Tai, S. C. Li, Y. N. Huang, C. M. Yu, and Y. S. Chiu, "Developments and applications of data deidentification technology under big data," *Journal of Electronic Science and Technology*, vol. 15, no. 3, pp. 231–239, 2017.

[3] X. Cheng, S. Su, S. Xu, and Z. Li, "DP-apriori: A differentially private frequent itemset mining algorithm based on transaction splitting," *Computers & Security*, vol. 50, pp. 74–90, 2015.

[4] C. Dwork, "Differential privacy," in *Lecture Notes in Computer Science*, pp. 1–12, 2006.

[5] C. Dwork, F. McSherry, K. Nissim, and A. Smith, "Calibrating noise to sensitivity in private data analysis," in *Theory of Cryptography Conference*, pp. 265–284, 2006.

[6] A. Ghosh, T. Roughgarden, and M. Sundararajan, "Universally utility-maximizing privacy mechanisms," *SIAM Journal on Computing*, vol. 41, no. 6, pp. 1673–1693, 2012.

[7] J. Han, M. Kamber, and J. Pei, "Data mining concepts and techniques third edition," *The Morgan Kaufmann Series in Data Management Systems*, vol. 5, no. 4, pp. 83–124, 2011.

[8] J. Han, J. Pei, Y. Yin, and R. Mao, "Mining frequent patterns without candidate generation: A frequent-pattern tree approach," *Data Mining and Knowledge Discovery*, vol. 8, no. 1, pp. 53–87, 2004.

[9] M. Kantarcioglu and C. Clifton, "Privacy-preserving distributed mining of association rules on horizontally partitioned data," *IEEE Transactions on Knowledge and Data Engineering*, vol. 16, no. 9, pp. 1026–1037, 2004.

[10] N. Li, W. Qardaji, D. Su, and J. Cao, "Privbasis: Frequent itemset mining with differential privacy," *Proceedings of the VLDB Endowment*, vol. 5, no. 11, 2012.

[11] S. Su, S. Xu, X. Cheng, Z. Li, and F. Yang, "Differentially private frequent itemset mining via transaction splitting," *IEEE Transactions on Knowledge*

and Data Engineering, vol. 27, no. 7, pp. 1875–1891, 2015.

[12] D. Wang and S. Long, "Boosting the accuracy of differentially private in weighted social networks," *Multimedia Tools and Applications*, vol. 78, no. 24, pp. 34801–34817, 2019.

[13] N. Wang, X. Xiao, Y. Yang, Z. Zhang, Y. Gu, and G. Yu, "Privsuper: A superset-first approach to frequent itemset mining under differential privacy," in *IEEE 33rd International Conference on Data Engineering (ICDE'17)*, pp. 809–820, 2017.

[14] T. Wang, N. Li, and S. Jha, "Locally differentially private frequent itemset mining," in *IEEE Symposium on Security and Privacy (SP'18)*, pp. 127–143, 2018.

[15] W. K. Wong, D. W. Cheung, E. Hung, B. Kao, and N. Mamoulis, "An audit environment for outsourcing of frequent itemset mining," *Proceedings of the VLDB Endowment*, vol. 2, no. 1, pp. 1162–1173, 2009.

[16] M. J. Zaki, "Scalable algorithms for association mining," *IEEE Transactions on Knowledge and Data Engineering*, vol. 12, no. 3, pp. 372–390, 2000.

[17] C. Zeng, J. F. Naughton, and J. Y. Cai, "On differentially private frequent itemset mining," *The VLDB Journal: Very Large Data Bases: A Publication of the VLDB Endowment*, vol. 6, no. 1, pp. 25, 2012.

Biography

Shigong Long is a Professor at the College of Computer Science and Technology at Guizhou University. He did his Ph.D. from the Institute of Computer Software and Theory of Guizhou University. His areas of interest include Cryptography, Formal Analysis of Software, and Privacy protection techniques. He has published 100 plus research papers in various journals and conferences of national and international repute.

Hongqin Lu is a master of computer science and technology in Guizhou University. His main research direction is the combination of differential privacy and mechanism design.

Tingting Chen has received the master's degree in computer science and technology from Guizhou University, China, in 2019. Her research interests include differential privacy and frequent itemsets mining.

Nannan Zhou is a master of computer science and technology in Guizhou University. Her main research direction is the combination of differential privacy and machine learning.

Hai Liu received his Ph.D. degree from Shanxi Normal University, China, in 2019, and his B.S. degree and M.S. degree from Guizhou University, China, in 2012 and 2015, respectively. He is now a PostDoc at Guizhou University. His main research interest includes privacy protection.

Research on Blackmail Virus Defense Based on Multi-Defender Hybrid Strategy

Guoqing Sun, Jiulun Fan, and Hao Wu
(Corresponding author: Guoqing Sun)

School of Cyberspace Security, Xi'an University of Posts and Telecommunications
Xi'an 710121, China[1]
Email:sgqxupt@163.com

(Received Dec. 17, 2020; Revised and Accepted June 6, 2021; First Online Nov. 10, 2021)

Abstract

Aiming at the problem of optimal strategy selection in blackmail virus security defence, starting from the motivation standpoint of both sides and considering the relationship between targets and defenders, the blackmail virus defence problem is modelled as a security game problem based on the mixed strategy of multiple defenders, A multi-threading algorithm is designed to solve the optimal strategy under the coordinated defence of defenders, Dynamic allocation of defence resources is realised. The experimental results show the effectiveness of the algorithm and analyze the impact of the target value and association changes on the defence strategy.

Keywords: Blackmail Virus; Multiple Defenders; Multi-threading; Optimal Strategy

1 Introduction

In recent years, the global spread and attack of the blackmail virus have caused considerable losses. Tencent security officially released the blackmail virus report in the first half of 2020, which shows that the blackmail virus is still very active in the first half of 2020, and more and more blackmail viruses are targeting the targets with higher data value and larger scale of traditional enterprises, education, medical care, government agencies, and so on. Taking an enterprise in Guangzhou, China, which was attacked by the Phobos blackmail virus, for example, after the blackmail virus invaded the first computer, it spread rapidly in the enterprise intranet. Through malicious tools, it constantly attacked the intranet machines, making more computers infected and unable to run normally. According to incomplete statistics, more than 20 servers of the enterprise were down after the attack, including two core database servers. Because the speed of the blackmail virus spreading and attacking is very fast, and the defender needs to consider the cost of statistical target information, the cost of communication between the target and the defender, and the time cost of dealing with the problem. It has not enough coordination ability and response time for a single defender to defend the blackmail virus. The most effective way is to increase the number of defenders to improve the coordination ability and shorten the response speed under the existing ability of the defender; By dispersing the defence target to multiple defenders, reducing the pressure of a single defender and improving the efficiency of it. Therefore, a more effective multi defender security game model has been proposed, and the way to coordinate the defenders to make the optimal defence strategy under the limited cost is also an urgent problem to be solved. In this paper, we use multi defender model to deal with the problem of blackmail virus defence, through the multi defender cooperative defence, control reasonable cost to complete the calculation of the optimal hybrid defence strategy.

The multi defender model needs to simulate the confrontation between the defender who protects the targets and the attacker who tries to destroy the targets and considers the correlation between the targets (defenders), to ensure that the defenders coordinate with each other and jointly protect all the targets [15]. The attacker always acts before the defender: After observing the defender's initial strategy, the attacker seeks the best attack target by maximising the attack expectation; The defender tries to obtain the maximum defence utility under the limited resources [3,6]. We need to design an optimal strategy selection algorithm based on multi defender security game after comprehensively analyzing the influence of defence cost and defence utility on strategy selection [1,5,8]. For the problem of optimal strategy selection in multi defender Game, [13, 19] considered the interaction between targets, formulated a game of one attacker and N defenders, and formed an optimisation problem for each participant to help select the optimal strategy. This paper also studies the security game of multiple defenders considering interdependence and proposes a mixed-integer linear programming formula to calculate the optimal strategy of multiple defenders [7]. Based on [7, 14] provides a more efficient optimal multi defender defence strategy selection

algorithm is designed by converting the optimal policy selection into a nonlinear programming problem.

In order to make decisions on multiple independent and possibly non-cooperative utility networks, and to ensure the maximum reduction of investment costs, a theoretical model of infrastructure network restoration based on the sequential game of multiple defenders based on interdependence is proposed in [12], which is expressed as the discrete-time non-cooperative game between networks. This paper studies the multi-player game between the intruder and the defender, considers the multi-player cooperative game and proposes a decomposition method to give a cooperative intrusion strategy to ensure the lower limit of profit [10]. However, in the case of virus defence, it is often to seek the maximum profit (upper limit) given the limited cost, which is also the starting point of our paper. In [16], a new allocation method is proposed to solve multiple defenders defending attackers. In other words, each defender is assigned to a unique attacker. The behaviour of each defender is determined and the upper limit of attacker's profit is given. In contrast, our paper presents and computes the defender's strategy (Non-fixed defence behaviour) through the hybrid strategy, making the defender's behaviour (strategy) more flexible. [11] Based on a novel and extensible (polynomial time) allocation algorithm, the algorithm can accommodate the cooperative behaviour between defenders, superior to the defence strategy without cooperative behavior. The security game including one attacker and multiple defenders considered in [18]is similar to our paper. A distributed defence strategy is proposed for the coordinated defence of defenders against attackers. However, in our paper, we use the distributed computing method to keep the game between each defender, and transfer the results of each thread game to other defenders immediately. Then, we use the correlation relationship to maintain the interaction between the defenders in decision-making to ensure the optimisation of the whole strategy. The model in [9] is an extension of the matrix attack and defence game model and Markov decision-making process and is a dynamic attack defence deduction model under multi defender model. However, it is mentioned in [17]that the defender is supposed to know the attacker's income in the matrix game, which is inconsistent with the actual situation, and an equilibrium solution method for the random game model with incomplete information is proposed.

The work of this paper has made three main contributions. In this paper, a multi defender security game model is established to solve blackmail defence. The best defence strategy of each defender is represented and calculated by using a mixed strategy. Different targets are divided into different defenders for management by using distributed computing, which avoids invalid strategies due to the insufficient ability of a single defender. It can ensure the mutual game between the defenders, dynamically coordinate the defence strategies among the defenders, and ensure the overall defence effectiveness to achieve the optimal. Finally, the experiment verifies the effectiveness of the algorithm. It gives the analysis of the influence of the parameter value on the strategy, which plays a guiding role in the selection of the optimal strategy in the blackmail virus defence problem.

Section 2 establishes the multi defender security game model under the mixed strategy; Section 3 gives the algorithm example; Section 4 analyzes the experimental results; Section 5 summarizes the experiment.

2 Multi Defender Security Game Model Under Mixed Strategy

2.1 Establishment of Attack And Defense Model

There are many defenders in the network $I = 1, 2, 3...n$, a single attacker, and a series of targets (nodes) T. Each defender i is responsible for protecting a set of targets $T_i = \{t_{i,1}, t_{i,2}, t_{i,3}, \cdots, t_{i,k}\}$, $T_i \subseteq T$ and target $j \epsilon T_i$.Now suppose $T_i \bigcap T_{i'} = \emptyset$ $(i \neq i')$ and $\bigcup_{i \in I} T_i = T$ (that is, the number of objects protected by each defender is more than 1, and the protection targets are not repeated between each other).

The strategy set of all defenders can be expressed as a mixed strategy matrix, as in Equation (1). Suppose that each defender sets corresponding protection strategies for each target he protects: We use probability $q_{i,j}$ $(q \in (0,1))$ to represent the security policy of defender i against target j; Therefore, the strategy of defender i against the target he protects should be a mixed set of policies, which can be expressed as $\{q_{i,t_{i,1}}, q_{i,t_{i,2}}, q_{i,t_{i,3}}, \cdots, q_{i,t_{i,k}}\}$ with $\Sigma_{k \epsilon T_i} q_{i,t_{i,k}} = 1$. On the other hand, assume that the attacker has limited resources and can only attack one target at a time. Then the target of the attacker is j.

$$q_{i,j} = \begin{bmatrix} q_{1,t_{1,1}} & q_{1,t_{1,2}} & q_{1,t_{1,3}} & \cdots & q_{1,t_{1,k}} \\ q_{2,t_{2,1}} & q_{2,t_{2,2}} & q_{2,t_{2,3}} & \cdots & q_{2,t_{2,k}} \\ q_{3,t_{3,1}} & q_{3,t_{3,2}} & q_{3,t_{3,3}} & \cdots & q_{3,t_{3,k}} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ q_{n,t_{4,1}} & q_{n,t_{4,2}} & q_{n,t_{4,3}} & \cdots & q_{n,t_{4,k}} \end{bmatrix} \quad (1)$$

As shown in Figure 1, the corresponding hybrid strategy matrix q is shown in Equation (2):

$$q_{i,j} = \begin{bmatrix} 0.5 & 0.5 & \\ 0.333333 & 0.333333 & 0.333333 \\ 0.333333 & 0.333333 & 0.333333 \end{bmatrix} \quad (2)$$

In the strategy matrix above, the first defender protects targets 3 and 7 with defence strategies of 0.5 the second defender protects targets 1, 4 and 8 with defence strategies of 0.333333333, 0.333333333 and 0.333333333; The third defender protects targets 2, 5 and 6, with defence strategies of 0.333333333, 0.333333333 and 0.333333333.

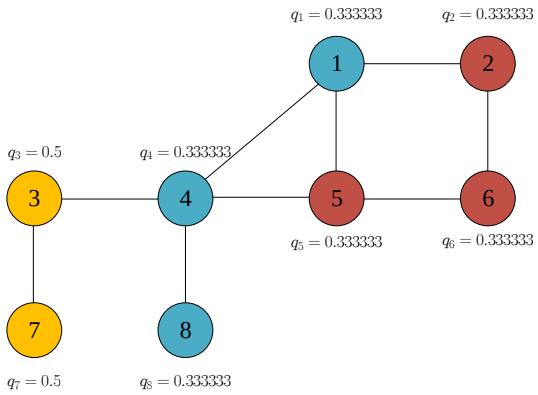


Figure 1: Defense strategy diagram

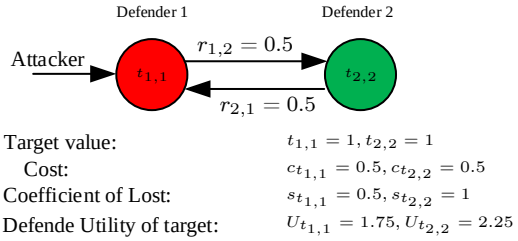


Figure 2: Schematic diagram of parameter values

2.2 Calculation of Defense Utility

Assuming that the defence effectiveness only depends on the target being attacked and the security strategy (defence cost) of the defender to the target, the attacker can only select one target with the greatest attack utility to attack each time. The value of each target j is recorded as v_j. if defender i protects target j, it will consume resource $c_{i,j}$. if target j is destroyed or infected, it will infect other targetj'. the possibility or influence of this infection is determined by "correlation degree $r_{j,j'}$ ". If the attacker attacks the target $j(j \in T)$, then the defender's utility value is recorded as $U_{i,j}$ and the attack utility value is recorded as $V_{i,j}$. We use Equation (3) to express defence utility:

$$U_{i,j} = (1 - s_j)(v_j + c_j + \sum_{j'}(v'_j + c'_j)r_{j,j'}). \quad (3)$$

Correlation degree r: The existence of association affects the effectiveness of attack and defence at the same time: The attack of blackmail virus starts from an initial target, and after the attacker causes direct damage to the target, it will also cause indirect damage to other targets due to the association relationship; After the defender allocates certain defence resources to the target and improves its defence effectiveness, it will indirectly increase the defence utility of other targets associated with this target. This indirect influence is transmitted by the correlation degree $r, r \in (0, 1)$, which $r_{i,j'}$ indicates the association degree between the current target and an associated target.

Defense investment c: The total defence resource consumed by all defenders is C, and the defence resource consumed by the defender i is $C_i, C = \Sigma_i C_i$. Moreover, the total defence cost C should also satisfy $C \leq C_{MAX} = \sum_i \sum_j v_{i,j}$. This is because if the defence investment cost $c_{i,j}$ of the target j is greater than the target value $v_{i,j}$, Then the defence strategy of the defender against the target is meaningless.

Loss coefficient s: Once an attack occurs, the parameter s_j $(s_j \in [0, 1])$ is used to reflect the effectiveness of the attacker's attack on target j, so s_j is called the loss coefficient of the attacked target j. When there is no attack in the system or the attacking object is not j, the value of s_j is 0; otherwise, the value of s_j will be defined according to the actual loss degree of the target being attacked (a value of 1 indicates that the target is completely lost after being attacked).

As shown in Figure (2), there are two defenders, each protecting a target separately, the value of each target is 1, the correlation degree of each target is 0.5, and the defence investment is 0.5; Assuming that the attacker chooses to attack the first target, the loss coefficient of the attacked target is 0.5; From Equation (3), the defence effectiveness of target 1 and target 2 are 1.75 and 2.25, respectively.

2.3 Calculation of Optimal Strategy

Because we use the hybrid strategy in this paper, when the defender deploys the defence strategy, the ultimate goal is to make the expectation of defence utility reach the optimal (maximum); The first step of strategy selection should determine the conditions for the system to achieve the optimal profit: That is, to maximise the expectation of defence utility of multiple defenders under a limited investment cost; The second step is based on the conditions of the optimal profit, propose a algorithm to calculate the optimal strategy; The maximization of defence utility expectation of multiple defenders means that the optimal profit problem of multiple defenders can be transformed into the optimal solution problem of optimization model.

On the one hand, the defence resources owned by each defender can only be used to deploy defence strategies for the targets under their control, so each defender is independent; On the other hand, the defenders are related because there is always a cascade relationship between boundary targets belonging to different defenders, and each defender has to take such additional effects into account. The multi defender algorithm proposed in this paper reduces the number of target variables and association complexity that each defender needs to consider by distributed computing [20]. At the same time, the attacker's changing strategy simulates the infection trend and effect of the virus (the process of the attacker making the attack strategy), and the game process between the defenders simulates the repeated calculation of the

defender in the real network environment, as well as the process of making the optimal defence strategy.
Algorithm steps:

1) n defenders protect all target sets T, and each defender i is responsible for protecting a group of targets $T_i = \{t_{i1}, t_{i2}, t_{i3}, \cdots, t_{ik}\}$, $T_i \subseteq T$, $n * k = m$, $j \in T_i$. Then the initial defence strategy set $q_{i,ini}[]$ of the defender i should satisfy uniform distribution, and the initial defence strategies are all $\frac{1}{k}$.

2) The defence investment C_i that one defender i can control is determined by the target it protects. According to the ratio of the total value of each defender's target, we can divide it into total defence investment C by Equation (4):

$$C_i = C\frac{\sum_j v_{i,j}}{\sum_i \sum_j v_{i,j}} \tag{4}$$

From Equation (4) above, we can get the defensive investment C_i of defender i, and then according to $c_{i,j} = q_{i,j}C_i$, we obtain the initial defence investment set $c_{i,j,ini}[]$ of defender i.

3) According to the defence investment set $c_{i,j,ini[]}$ and substituting into Equation (3), the defense utility set $U_{i,ini}[]$ is obtained.

4) The attack utility is numerically equal to the defence utility of the target, And the best target of the attacker is often the one with the greatest attack effect, Therefore, the maximum attack utility can be obtained by Equation (5):

$$V_{max} = \max V_{i,j} = |-\max U_{i,j}| \tag{5}$$

Then the initial attack utility set $V_{ini}[]$, the initial maximum attack utility $V_{max,ini}$, the initial optimal attack target j_{ini} and the initial loss coefficient set $s_{i,ini}[]$are calculated by the above equation;

5) Bring the uniformly distributed initial response value $q_{i,j,ini}[]$ into Equations (6), (7), (8), to calculate the new protection strategy $q_{i,j,Iter}[]$ in defender i:

$$max_{q,c,U} \sum_j q_{i,j,Iter}U_{i,j,ini} - \sum_j q_{i,j,Iter}C_{i,j,ini} \tag{6}$$

s.t

$$0 \le q_{i,j,Iter} \le 1, \forall j \in T_i \tag{7}$$

$$\sum_{j \in T_i} q_{i,j,Iter} = 1, \forall j \in T_i \tag{8}$$

6) Other defenders start with the initial defender i, according to their own order in the infectious path starting from i, and the latter defender i^* is based on the previous calculation results of defender i (Optimal defence strategy set and defence utility set), the association relationship with the former, repeat the above steps until all defenders complete solving their own defence strategy set, defence investment set and defence utility set.

7) The above steps demonstrate that a defender has completed the process of solving the defence strategy. Before the iteration stop condition is reached, we get the new optimal attack target according to Equations (1), (3), and iterate continuously according to the new optimal attack target to solve the solution of the system optimal defence strategy satisfying the iterative conditions. The above calculation process is shown in the following pseudo code (Algorithm 1).

Algorithm 1 Optimal strategy calculation in multi defender game

```
Begin
Initialize the Target value set v[], Association degree set r[], Initial defense strategy set q[], Total defense investment C, Number of iterations t, Weighted coefficient set of initial defense loss s[].
Calculate Initial defense utility set U_{i,j} = (1−s_j)(v_j + c_j + Σ_{j'}(v'_j + c'_j)r_{j,j'}) Initial defense investment set c_{i,j}[] = q_{i,j} × C (Σ_j v_{i,j}) / (Σ_i Σ_j v_{i,j}) tmp[] = U[].
while 1 ≤Iterate times≤ t do
    Compute Attack target j
    Set of defense loss weighting coefficients s[] by V_j = maxtmp[];
    U[] = s_j[]U[];
    while 1 ≤Number of defender a ≤ t do
        Running Thread i...
        Compute Defense strategy of defender i:
        max Σ_j q_{i,j}U_{i,j} − Σ_j q_{i,j}c_j
        Update Defensive utility of defender a:
        U_{i,j} = (1 − s_j)(v_j + c_j + Σ_{j'}(v'_j + c'_j)r_{j,j'})
    end while
    return q_i[] and U_i[]
end while
return q[] and U[]
End
```

3 Algorithm Example

An example is given in this section, as shown in Figure 3: Three departments manage eight computers, of which department A protects the yellow target computers (targets 3 and 7), Department B protects the red target computers (targets 1, 4, 8), and department C protects the blue target computers (targets 2, 5, 6). The value of each target is set to 1, and the iteration times of the algorithm are 1000. When the attacker attacks the target, the loss coefficient s is 0.7 (the loss coefficient can be arbitrarily selected in the interval $[0, 1]$).

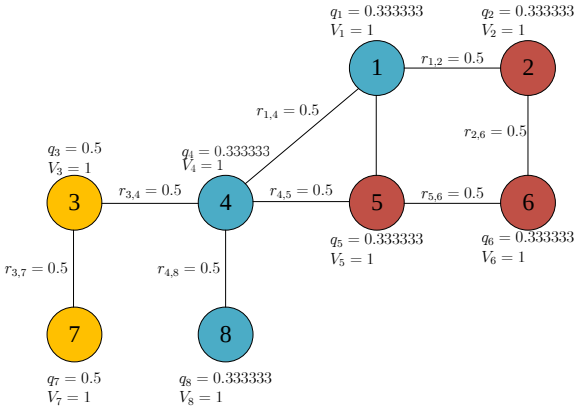


Figure 3: Target value and initial strategy

In order to resist the attack of blackmail virus on the internal equipment and data of the company, the company has invested a total of defence costs, which are used to train employees' security skills, upgrade system patches, update firewall configuration, strictly manage ports, backup important data in different places and multiple points. Now, the company has given the total defence investment $C = 4$. However, the company will not allocate the specific defence investment of each target computer in detail; Department i reports the number of computers and the value of data to the company, and the defence investment allocated by the company to department i is C_i, and the defence investment $c_{i,j}$ allocated by each computer is arranged by its department i:

1) According to Equation (4), the disposable defence investment can be divided into three parts:

$$C_{A,ini} = 1, C_{B,ini} = 1, C_{C,ini} = 1. \tag{9}$$

2) According to the topology of the network and the division of the target, we give the initial defence strategy as Equation (10), The initial strategy is uniformly distributed:

$$q_{i,j} = \begin{bmatrix} 0.5 & 0.5 & \\ 0.333333 & 0.333333 & 0.333333 \\ 0.333333 & 0.333333 & 0.333333 \end{bmatrix} \tag{10}$$

3) According to $c_{i,j} = q_{i,j}C_i$, the defence investment of each computer in each department is shown in Equation (11). It can be seen that the initial strategy satisfies the uniform distribution:

$$c_{ini} = \begin{bmatrix} 0.5 & 0.5 & \\ 0.5 & 0.5 & 0.5 \\ 0.5 & 0.5 & 0.5 \end{bmatrix} \tag{11}$$

4) Each department calculates the defence utility set by substituting the initial defence investment set into Equation (12):

$$U_{ini} = \begin{bmatrix} 3.0 & 2.25 & \\ 3.75 & 4.5 & 2.25 \\ 3.0 & 3.75 & 3.0 \end{bmatrix} \tag{12}$$

5) Select the target computer with the largest attack effect by Equation (3), the target 4 computer is the best target for attacker in the first round.

6) Each defender substitutes his initial defense strategy into Equations (6), (7), (8), to obtain the defence strategy set after the first iteration of departments A, B and C:

$$q_{Iter-1} = \begin{bmatrix} 0.800781 & 0.199219 & \\ 0.932292 & 0 & 0.067708 \\ 0.218575 & 0.609933 & 0.171492 \end{bmatrix} \tag{13}$$

7) After getting the new defence strategy, we update the investment set according to $c_{i,j} = q_{i,j}C_i$, and then according to Equation (3), updating the defence utility set (Equation (14)):

$$U_{Iter-1} = \begin{bmatrix} 2.900391 & 2.099609 & \\ 4.494792 & 0.461654 & 1.601562 \\ 3.155701 & 4.242737 & 2.878619 \end{bmatrix} \tag{14}$$

8) To the last step, the algorithm has completed an optimal strategy calculation and got the result of the initial iteration, but this result hasn't considering the change of attacker's strategy and the game between defenders. Therefore, we need to iterate the algorithm to deal with these two problems; The new maximum attack utility $V^{**}_{max} = V_3 = 4.494792$ is obtained from Equation (5). At this time, if the iteration stop condition is not reached, the third target corresponding to V^{**}_{max} is regarded as the new attack target, and Steps (6) and (7) are repeated to continue the next optimal strategy calculation until the iteration stop condition is reached.

4 Analysis of Experimental Results

Through the game between attacker and defender, as well as between defender and defender, the following optimal defence strategy matrix (shown in Equation (15)) and defence utility matrix (shown in Equation (16)) are obtained. The distribution of defence strategy in the network is shown in Figure 4:

$$q_{Iter-1} = \begin{bmatrix} 0.670513 & 0.329487 & \\ 0 & 1 & 0 \\ 0.085805 & 0.696038 & 0.218156 \end{bmatrix} \tag{15}$$

$$U_{Iter-1} = \begin{bmatrix} 3.585256 & 2.164744 & \\ 0.389172 & 5.301573 & 2.250000 \\ 2.292326 & 4.457674 & 2.913617 \end{bmatrix} \tag{16}$$

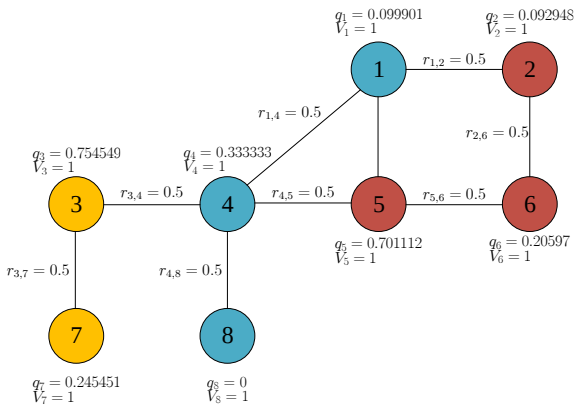


Figure 4: Defence Strategy after game

We compare the full-game defence strategy with the initial state, the result is shown in Table 1: After considering the mutual game between defenders, the total defence utility will be increased by 2.57% compared with the initial state; The change of strategy of each target is shown in Table 1. Compared with the results of the game, in the initial state, defender A tilts too many defence resource to target 3; Defender B places much defence resource on target 1, which fails to protect core target 4; And defender C's strategy changes little. By reducing the defence strategy of target 3 and improving the defence strategy of target 7, the game appropriately balances the strategy allocation gap between the two targets in defender A; While the defence resources of defence B incline to target 4; This is because target 4 is in the "core position of the network" (we call the target with the most complex relationship as the core target, and the area with the most dense distribution of core targets is called the core area) The resource of defender C inclines to target 5 and target 6 after iteration because target 5 and target 6 are closer to the core area of network than target 2. It is not difficult for us to find that after the defence strategy is adjusted to the target (target set) in the core position (area) of the network, the defender has achieved greater defence effectiveness.

4.1 The Impact of the Change of Target Node Value

Due to the existence of association, if the value of a node in the network changes, the defence strategy of itself and other adjacent targets will change too. Therefore, we take the modification of target 4 in defender C as an example, reduce the value of target 4 to 90% of the original value, that is, $v_4 = 0.1$.The following optimal defence strategy matrix (Equation (17)) and defence utility matrix (Equation (18)) are obtained. The distribution of defence strategy in the network is shown in Figure 5:

$$q_{Iter-fin-v} = \begin{bmatrix} 0.520009 & 0.479991 & \\ 0 & 0.873650 & 0.126354 \\ 0.215490 & 0.496183 & 0.288327 \end{bmatrix} \quad (17)$$

$$U_{Iter-fin-v} = \begin{bmatrix} 2.979019 & 2.239960 & \\ 0.329826 & 4.092519 & 1.894766 \\ 2.539480 & 3.665755 & 2.966245 \end{bmatrix} \quad (18)$$

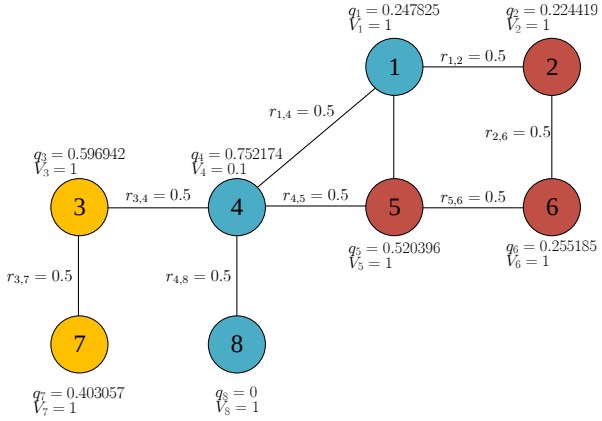


Figure 5: Defence Strategy after target 4 value is reduced by 90%

After the value of target 4 is reduced, the total defence utility decreases by 11.27% with the iteration (game) going on; The defence resource given by defender B to target 4 are reduced, but because it is in the core position, the change rate of resource is not big. The reduced part resource is added to target 1, and target 8 is abandoned because it is at the edge of the network; Target 3 in defender A is directly related to target 4,so the defence resource of target 3 will decrease with target 4 as well, and the reduced part of defence resource will be increased to target 7; Similarly, target 5 in defender C is also directly related to target 4, so the defence strategy given by defender C to target 5 is also reduced, and the corresponding reduced part of defence resource is increased to target 2 and target 6. This result shows that if the value of a target (area) in the network decreases, the defender will reduce the defence resource of this target (area), and the reduced part of defence resource will shift to another target (area).

4.2 The Influence of The Change The Target Node Association Degree

Then we directly modify the association between target 4 and 5, reducing the association by 80%, that is $r_{4,5} = 0.1$, The following optimal defence strategy matrix (Equation (19)) and defence utility matrix (Equation (20)) are

Table 1: Comparison of the results of the game between defenders

Data	target 3	target 7	target 1	target 4	target 8	target 2	target 5	target 6	Total defence utility
Initial state policy	0.80078	0.19922	0.65354	0.34646	0	0.16714	0.65514	0.17773	24.925456
Strategy under the game	0.75455	0.24545	0.09999	0.90010	0	0.09295	0.70111	0.25094	25.565008
Strategy change	-0.04623	0.04623	-0.55364	0.55364	0	0.07419	0.04598	-0.02822	0.639552
Strategy change rate	-5.77%	23.21%	-84.71%	23.21%	-%	-44.39%	7.02%	15.88%	2.57%
Defence utility distribution	3.478543	2.122725	3.948872	5.295105	2.175074	2.368803	4.456124	2.904455	

Table 2: Comparison of experimental results before and after reducing the value of goal 4 by 90%

Data	target 3	target 7	target 1	target 4	target 8	target 2	target 5	target 6	Total defence utility
Strategy under v_4=0.1	0.59694	0.40306	0.24783	0.75217	0	0.22442	0.52040	0.25519	22.683588
Strategy under the game	0.75455	0.24545	0.09999	0.90010	0	0.09295	0.70111	0.25094	25.565008
Strategy change	-0.15761	0.15761	0.14792	-0.14792	0	0.13147	-0.18072	0.04925	-2.881142
Strategy change rate	-20.89%	64.21%	148.07%	-16.43%	-%	141.45%	-25.78%	23.91%	-11.27
Defence utility distribution	2.858104	2.201529	3.567352	4.08542	1.614131	2.713886	3.771983	2.941389	

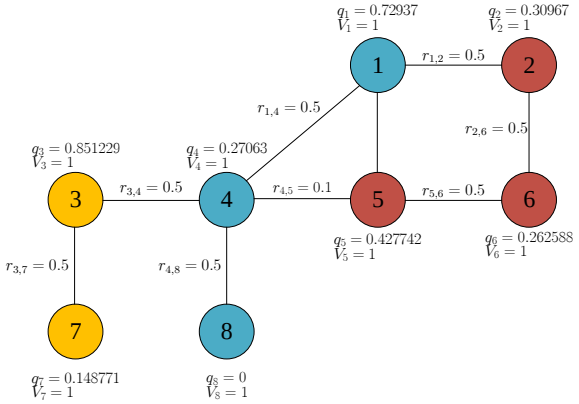


Figure 6: Defence strategy with 80% reduction of correlation degree of targets 4 and 5

obtained. The distribution of defence strategy in the network is shown in Figure 6:

$$q_{Iter-fin-r} = \begin{bmatrix} 0.668469 & 0.331531 & \\ 0 & 0.842929 & 0.157061 \\ 0.274336 & 0.385642 & 0.340022 \end{bmatrix} \quad (19)$$

$$U_{Iter-fin-r} = \begin{bmatrix} 3.487053 & 2.165765 & \\ 0.373682 & 4.382069 & 2.367796 \\ 2.666521 & 3.059920 & 3.005016 \end{bmatrix} \quad (20)$$

After reducing the correlation degree between targets 4 and 5, the overall defence effectiveness is reduced by 6.99%; The defence strategies of targets 4 and 5 are reduced by 69.93% and 38.99%, respectively, which indicates that the reduction of association will also reduce the defence resource of the corresponding two targets; From the perspective of target 4 in defender B, those targets are related to it directly: Targets 3 in defender A, target 5 in defender C and target 1 and target 8 in defender B. except that target 5 reduces the defence resource due to its association with target 4, defender A inclines the defence resource to target 3, and defender B inclines the defence resource to target 1. Target 5 in defender C is associated with target 1, target 4 and target 6 in defender B. Except that target 4 of the defence strategy is reduced due to the same correlation, Defender B increases defence resource of target 1 and defender C increases defence resource of target 6. This result shows that in the interconnected network, when the correlation between two targets is reduced, the defence resource and defence effectiveness of these targets will also decrease, and the defender will improve the defence resource of other adjacent targets to make up for the loss.

4.3 The Distribution of Defence Effectiveness

In this section, we try to analyze the defence utility distribution. The defence utility distribution in the above experimental results is shown in Figures 7, 8and 9:

After comparing the different experimental results' defence effectiveness distribution, we find that whether it is the change of target value or the change of target correlation relationship, the final defence resources of defender A, B and C always incline to the "core area". This is because the targets in the core area tend to have more complex correlation relationship or higher target value. Therefore, the defence resources constructed in this paper are designed in the imperial model, the method to approach the optimal overall defence effectiveness is: All the defenders' overall strategies should try their best to ensure the safety of the targets in the "core" area, so as to obtain the maximum defence effectiveness. For targets in non-core areas, such as target 8,for example, Based on this paper, we can build honeypot system by adding network deception game model and using network deception technology to obtain greater defence effectiveness and achieve better defence effect [2, 4].

Table 3: Comparison of experimental results before and after reducing the correlation of $R_{4,5}$ by 80%

Data	target 3	target 7	target 1	target 4	target 8	target 2	target 5	target 6	Total defence utility
Strategy under $r_{4,5}$=0.1	0.85123	0.14877	0.72937	0.27063	0	0.30967	0.42774	0.26259	23.777021
Strategy under the game	0.75455	0.24545	0.09999	0.90010	0	0.09295	0.70111	0.25094	25.565008
Strategy change	0.09668	-0.09668	0.62947	-0.62947	0	0.21672	-0.27337	0.05665	-1.787987
Strategy change rate	12.81%	-39.39%	630.09%	-69.93%	-%	233.16%	-38.99%	27.51%	-6.99%
Defence utility distribution	3.18409	2.07439	4.30624	4.03963	1.70297	3.20847	3.52618	2.94694	

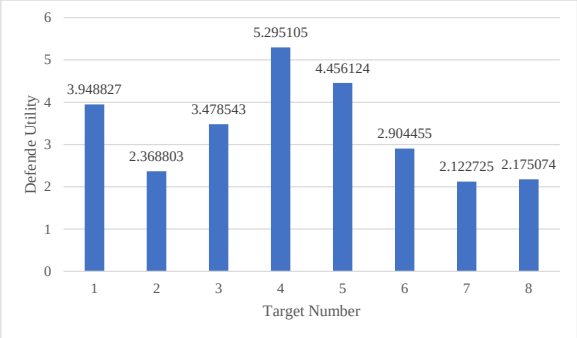


Figure 7: The distribution of defence utility in the game between defenders

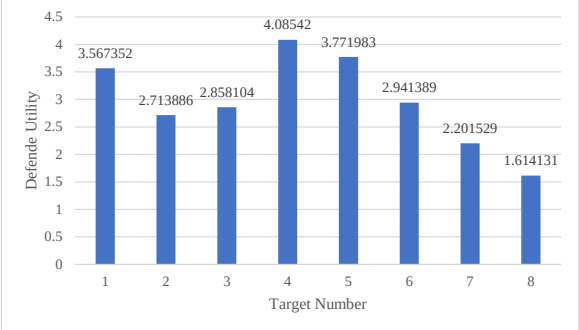


Figure 8: Defence utility distribution after target 4 value is reduced by 90%

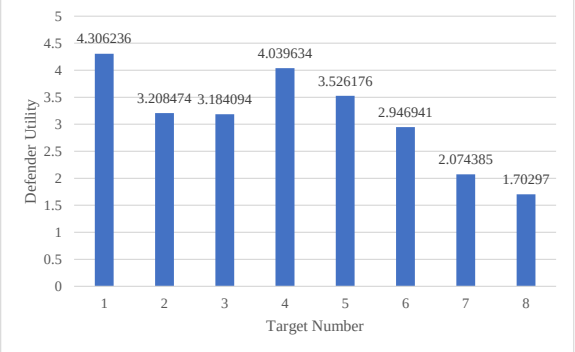


Figure 9: Defensive utility distribution after 80% reduction of correlation degree of targets 4 and 5

5 Conclusion

Aiming at the problem of selecting the best strategy in blackmail virus defence, this paper proposes a distributed algorithm based on the multi defender security game model of hybrid strategy, which is used to solve the optimal defence strategy of the whole defender. The algorithm maximizes the expectation of defence profit and ensures the coordinated defence among defenders. In the experiment of changing the target value or association relationship of the network, the distribution of defence strategy always tends to the area where the "network core" is located. The above experiments show the effectiveness of the algorithm, and play a guiding role in the strategy selection when defending the blackmail virus. Because this model does not consider the situation that multiple targets are attacked at the same time, in the next step, multiple attackers will be added to make the attacker change from single point attack to multi-point cooperative attack, and change the attacker's strategy, which is represented and calculated by mixed attack strategy.

Acknowledgments

This study was supported by the National Key R&D Program of China(Grant No.2017YFC0803800); NSFC funded project (Grant No.61671377).

References

[1] X. Chen, X. Liu, L. Zhang, and C. Tang, "Optimal defense strategy selection for spear-phishing attack based on a multistage signaling game," *IEEE Access*, vol. 7, pp. 19907–19921, 2019.

[2] Y. Guo, H. Zhang, Y. Hu, J. Ma, "Research on active defense based on multi-stage cyber deception game," *Journal on Communications*, vol. 41, no. 8, pp. 32, 2020.

[3] D. Guerrero, A. A. Carsteanu, R. Huerta, and J. B. Clempner, "An iterative method for solving stackelberg security games: A markov games approach," in *The 14th International Conference on Electrical Engineering, Computing Science and Automatic Control (CCE'17)*, pp. 1–6, 2017.

[4] H. Hu, Y. Liu, H. Zhang, and R. Pan, "Optimal network defense strategy selection based on incomplete information evolutionary game," *IEEE Access*, vol. 6, pp. 29806–29821, 2018.

[5] C. Lei, D. H. Ma, and H. Q. Zhang, "Optimal strategy selection for moving target defense based on markov game," *IEEE Access*, vol. 5, pp. 156–169, 2017.

[6] G. Levitin, "Optimal defense strategy against intentional attacks," *IEEE Transactions on Reliability*, vol. 56, no. 1, pp. 148–157, 2007.

[7] J. Lou, A. M. Smith, and Y. Vorobeychik, "Multi-defender security games," *IEEE Intelligent Systems*, vol. 32, no. 1, pp. 50–60, 2017.

[8] J. Lou and Y. Vorobeychik, "Equilibrium analysis of multi-defender security games," in *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence (IJCAI'15)*, vol. 15, pp. 596–602, 2015.

[9] C. X. Shi and G. H. Yang, "Distributed nash equilibrium computation in aggregative games: An event-triggered algorithm," *Information Sciences*, vol. 489, pp. 289–302, 2019.

[10] D. Shishika and V. Kumar, "Local-game decomposition for multiplayer perimeter-defense problem," in *IEEE Conference on Decision and Control (CDC'18)*, pp. 2093–2100, 2018.

[11] D. Shishika, J. Paulos, and V. Kumar, "Cooperative team strategies for multi-player perimeter-defense games," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 2738–2745, 2020.

[12] A. M. Smith, A. D. González, L. Dueñas-Osorio, and R. M. D'Souza, "Interdependent network recovery games," *Risk Analysis*, vol. 40, no. 1, pp. 134–152, 2020.

[13] A. Smith, Y. Vorobeychik, and J. Letchford, "Multidefender security games on networks," *ACM SIGMETRICS Performance Evaluation Review*, vol. 41, no. 4, pp. 4–7, 2014.

[14] J. Tan, C. Lei, H. Zhang, and Y. Cheng, "Optimal strategy selection approach to moving target defense based on markov robust game," *Computers & Security*, vol. 85, pp. 63–76, 2019.

[15] M. Waniek, T. P. Michalak, and A. Alshamsi, "Strategic attack & defense in security diffusion games," *ACM Transactions on Intelligent Systems and Technology (TIST'19)*, vol. 11, no. 1, pp. 1–35, 2019.

[16] S. Yang, W. Xie, C. Wu, W. Wang, and Y. Zhang, "A hybrid model for optimal defense strategy generation," in *The 28th International Conference on Computer Communication and Networks (ICCCN'19)*, pp. 1–6, 2019.

[17] J. N. Yang, H. Q. Zhang, and C. F. Zhang, "Defense decision-making method based on incomplete information stochastic game," *Chinese Journal of Network and Information Security*, vol. 4, no. 8, pp. 12–20, 2018.

[18] J. Yin and M. Ye, "Distributed nash equilibrium computation for mixed-order multi-player games," in *IEEE 16th International Conference on Control & Automation (ICCA'20)*, pp. 1085–1090, 2020.

[19] M. Yu and S. H. Hong, "A real-time demand-response algorithm for smart grids: A stackelberg game approach," *IEEE Transactions on Smart Grid*, vol. 7, no. 2, pp. 879–888, 2015.

[20] H. W. Zhang, D. Yu, J. Hang, *et al.*, "Defense policies selection method based on attack-defense signaling game model," *Journal on Communications*, vol. 37, no. 5, pp. 51–61, 2016.

Biography

Guoqing Sun (1996 -), male, from Ziyang, Sichuan Province, He is currently pursuing his master's degree in information security,Xi'an University of Posts and telecommunications.His research interests is information security.

Jiulun Fan (1964 -), male, from Wenxian County, Henan Province, Ph.D,Professor of Xi'an University of Posts and telecommunications. Processing and information security.

Hao Wu (1981 -), male, from Wujin, Jiangsu Province, lecturer of Xi'an University of Posts and telecommunications, mainly focuses on information security.

Publicly Verifiable Outsourcing Computation for Inner Product Evaluation under Multiple Keys with Improved Security and Efficiency

Zhiqiang Du[1], Dong Zheng[1,2], and Qinglan Zhao[1]
(Corresponding author: Qinglan Zhao)
National Engineering Laboratory for Wireless Security, Xi'an University of Posts and Telecommunications[1]
Xi'an 710121, China
Email: zhaoqinglan@foxmail.com
Westone Cryptologic Research Center, China[2]
(Received Dec. 7, 2020; Revised and Accepted May 6, 2021; First Online Nov. 10, 2021)

Abstract

In an IoT network, sending the big data streams to the cloud for group-by-sum and the inner product has become a basic technique. There are two aspects to be considered. First, due to the indeterminacy of the cloud, security and efficiency have become two essential indicators to measure an outsourcing scheme. Second, it is practical that the scheme should support any two data sources to perform inner product, and each data source should be equipped with a unique key, which can help clients trace back to the original data source. In this paper, firstly, we find there are still some security flaws in the previous works. Concretely, the tags can be forged, and the input data can be replaced. Next, we focus on the group-by sum and inner product under the multi-keys and propose two publicly verifiable schemes. It is worth mentioning that our schemes have no security and are secure under the co-CDH assumption. Finally, the experiment evaluation illustrates the efficiency of our scheme.

Keywords: Cloud Outsourcing; Group-By Sum; Inner Product; IoT; Multiple Keys; Publicly Verifiable Computation

1 Introduction

Because of the advancement of wireless sensor devices [7], the IoT technology has developed rapidly and it continues to integrate with other new technologies, such as artificial intelligence, block chain. Consequently, the gap between the size of data streams which are generated by devices and the size of data streams which the clients can handle is increasing. Thus, uploading the messages to a dedicated server for analysis, aggregation, storage has become a natural processing mode [19].

Cloud computing also has some risks, the cloud may wilfully returns an arbitrary result for own savings or a malicious value due to the benefits which are from the competitors. Hence, we must ensure that the final result is correct, and the designed scheme can detect any malicious behavior from cloud. Currently, supporting the public verification has become a useful solution. With this property, clients can authenticate by themselves or delegate authentication to a trusted third party [4]. Moreover, when clients run an outsourcing algorithm, it is necessary to ensure that the cost of clients is less than original cost, and the cost is preferably constant [12].

Group-by sum and inner product, which are two fundamental operations in IOT, have many practical applications, such as data mining. There are two classes of schemes. One class are schemes with single-key, in which data is from one data source [1, 3, 4, 10, 13, 14, 21]. The other are classes with multiple-keys, in which data is from multiple data sources [2, 12, 16]. However, in some outsourcing scenarios which are based on the IOT, schemes with multi-keys are more practical. For example, in market analysis, each market uploads their own data to the regulatory authority for statistical analysis. And further more, we hope that the key of each data source is different. This method ensures that clients can trace back to the data source by unique key [12] if they are object to the results with the cloud honestly running the algorithm.

1.1 Related Work

The traditional outsourcing model has only two parties, cloud and clients. The main process is that clients outsource a complex computing task to the cloud, and this task can be a scientific computation with high complexity, such as matrix multiplication, matrix inversion, linear programming, determinant, linear regression [8, 9, 11, 20, 25] or an expensive cryptographic operation, such as modular inverse, bilinear pairings and modular exponentiations [15, 17, 18]. In [8] and [9], the effi-

cient freivalds' algorithm is used for verification, but it is a probabilistic verification algorithm. Therefore, in order to improve security, people have studied public verification [3, 4, 22, 23]. Compared with the traditional model, their models added a trusted verifier for public verification. The schemes in [3] and [23] also support public delegation.

Single-key: Nath *et al.* [13] proposed the verifiable grouped aggregation query, and their scheme supports public verification. Papadopoulos *et al.* [14] researched three calculations, group-by sum, inner product, matrix product under multiple data sources, their schemes only support private verifiability. Backes *et al.* [1] used full homomorphic encryption technology for data operation. Fiore *et al.* [4] presented a scheme of matrix multiplication with public verification. Then Elkhiyaoui *et al.* [3] improved the fiore's scheme and made it more efficient. However, both [3] and [4] require a matrix to be determined in advance. Zhang *et al.* [21] proposed two schemes for inner product and matrix multiplication (EPPDMM). Zheng *et al.* [24] found cloud could replace data maliciously in EPPDMM. To improve the efficiency of EPPDMM, Zhang *et al.* [22] also proposed two efficient schemes POMM and AFEMM for matrix multiplication. We find there is still a similar security flaw to EPPDMM. Li *et al.* [10] presented a securely and publicly verifiable aggregation scheme in IOT. Zheng *et al.* [24] improved the EPPDMM and proposed two schemes for inner product and matrix multiplication. While a vector or a matrix needs to be determined in [10] and [24].

Multi-keys: The full homomorphic encryption technology was used in [2]. Liu *et al.* [12] achieved three publicly verifiable schemes for group-by sum, inner product, matrix product under multi-keys with very attractive efficiency. But Wang *et al.* [16] proved that Liu's [12] works are not secure, they found tags can be forged by adversary in [12] and proposed two modified schemes.

1.2 Our Contributions

After analyzing the schemes in Wang's [16], we find the correct tags can still be easily forged for arbitrary data in Wang's first modified scheme, and this scheme is only suitable for group-by sum query. Subsequently, in their second modified scheme, they just transformed the order of groups from prime order into the composite order. However, in composite order, the computation cost of group and pairing will be very high [5]. Meanwhile, we find cloud can replace the current input data by using previous data in these two schemes [12,16]. Consequently, for the outsourcing of group-by sum and inner product, there is still some room to improve.

Here, for group-by sum and inner product, we present two outsourcing schemes which support public verification, and the second scheme for inner product is based on the first scheme for group-by sum. With some preprocessing works, we have improved security and efficiency. In the mean time, due to the pre-computation, our scheme is mainly applied to the scenario which the dimension of source data is fixed. Then data sources can compute some public keys and system parameters under the amortized model [6]. our contributions are represented as the following four pionts.

1) For the works of [12, 16], we find there are still two security flaws, adversary can forge the correct tags for arbitrary data streams, and cloud can replace the current input data by using previous data.

2) In our two secure schemes, we construct a new tags.

3) In security analysis, we show our schemes which have no security flaws. Moreover, under the co-CDH assumption, we prove that our works are secure.

4) Experiment evaluation shows the efficiency at the side of data source and clients.

1.3 Organization

The other works are achieved as below: System model, algorithm framework are defined in Section 2. Section 3 gives the related definitions. Our attacks on previous works are given in Section 4. Following, Section 5 shows the detailed schemes. Subsequently, the security analysis is described in Section 6. We also provide performance evaluation and experiment evaluation in Section 7 and Section 8. Ultimately, the conclusion is presented in Section 9.

2 Scheme Formulation

2.1 System Model

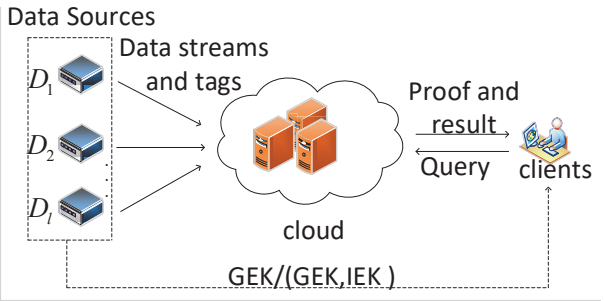


Figure 1: System model

Our model is showed in Figure 1. We use l denotes the total number of data sources and n denotes the dimension of original information. Let data source D_j ($j \in [1, l]$) collect the information $\gamma_j = \{\gamma_{j,1}, \gamma_{j,2}, ..., \gamma_{j,n}\}$ and send them to cloud. For more practical purposes, we consider that the data sources cannot be communicated with each

other. Here, we only focus on the efficiency and security, and data privacy is not be discussed. The following content is the detailed process of the outsourcing scheme.

Data sources: Each data source continuously collects the specified information and generates data in the form of stream. It can be an intelligent terminal in the IoT or an entity that can produce data streams, such as a large supermarket. Firstly, it has same public system parameters. Next, in each data source, it has a uniquely public key and a secret key. Then it calculates evaluation key GEK and returns it to the clients (In inner product, it also needs two publicly auxiliary keys, AK_j, AK'_j and another evaluation key IEK). Subsequently, it generates publicly verifiable tags $\rho_j = \{\rho_{j,1}, \rho_{j,2}, ..., \rho_{j,n}\}$. At last, it sends data streams and tags to the cloud.

Cloud: We deem that the cloud is malicious. After receiving the request, cloud performs the operation of group-by sum or inner product. At the same time, it generates proofs to prove the correctness of result. Finally, cloud returns them to the clients.

Clients: We consider that the clients are also malicious and they can not get the secret key from data sources directly. When clients receive the result and proofs, they use public messages to check whether the result is correct.

2.2 Algorithm Framework

Our publicly verifiable schemes include four modules (KeyGen, GenTag, Compute, Verify). $\gamma_j = \{\gamma_{j,1}, \gamma_{j,2}, ..., \gamma_{j,n}\}$ and $\gamma_k = \{\gamma_{k,1}, \gamma_{k,2}, ..., \gamma_{k,n}\}$ denote the data streams which are produced by D_j and D_k. Let f_1 denote group-by sum and f_2 denote inner product separately. For f_1, the query is $ret = \sum_{i=1}^{n} \gamma_{j,i}$. And for f_2, the query is $ret_1 = \sum_{i=1}^{n} \gamma_{j,i} \cdot \gamma_{k,i}$.

1) **KeyGen**$(f_1, 1^\kappa, t) \to (b_j, PK_j, GEK, AK_j)$: For f_1, the unique key is computed in this phase. After inputting the security parameter κ and query times t, D_j outputs a secret key b_j. Next it produces a public key PK_j. Different from previous works [12], [16], we also set a evaluation key GEK. For f_2, besides the works in group-by sum, D_j also needs to compute publicly auxiliary keys AK_j, AK'_j and another evaluation key IEK. That can be described as

$$(f_2, 1^\kappa, t) \to (b_j, PK_j, GEK, IEK, AK_j, AK'_j)$$

All keys can be computed in offline stage.

2) **GenTag** $(f_{1/2}, \gamma_{j,i}) \to (\rho_{j,i})$: The public tags are computed in this phase. D_j outputs public tags $\rho_{j,i}$ by source data $\gamma_{j,i}$ for f_1 and f_2.

3) **Compute** $(f_1, \rho_j, \gamma_j) \to (ret, \Gamma)$: This phase is run by cloud. For f_1, cloud produces result ret and proof Γ by inputting ρ_j and γ_j. For f_2, cloud outputs ret_1, ret_2 and proofs $\Gamma_1, \Gamma_2, \Gamma_3, \Gamma_4$ by the input tuple $(f_2, \rho_j, \rho_k, AK_j, AK'_j, \gamma_j, \gamma_k)$. That is $(f_2, \rho_j, \rho_k, AK_j, AK'_j, \gamma_j, \gamma_k) \to (ret_1, ret_2, \Gamma_1, \Gamma_2, \Gamma_3, \Gamma_4)$.

4) **Verify** $(f_1, GEK, PK_j, \varepsilon, ret, \Gamma)/(f_2, GEK, IEK, \lambda_1, PK_j, PK_k, ret_1, ret_2, \Gamma_1, \Gamma_2, \Gamma_3, \Gamma_4) \to$ (*accept 1 or reject 0*): For clients, utilizing the public messages, they ascertain $ret = f_{1/2}(\gamma_j)$. If ret is correct, output 1. If not, output 0.

3 Preliminaries

3.1 Security Definition

We have our experiment $Exp_A^{1^k}$ under the adaptive chosen-message attack for f_1. For f_2, the experiment is similar to f_1.

Step 1: The challenger B generates public/secret key and public messages for data source D_j. Then it shows public keys to adversary A.

Step 2: Adaptively, adversary A requests the challenger B to generate the public tags for the chosen messages on the discrete time. If B receives the challenge tuple (D_j, γ_j), it will generate tags ρ_j and return them to A.

Step 3: According to the challenge input (D_j, γ_j), adversary A gives (ret^*, Γ) to B with $ret^* \neq f_1(\gamma_j)$.

Step 4: If $verify(ret^*, \Gamma) = 1$, we get adversary A succeeds in $Exp_A^{1^k}$.

Definition 1. *Let Pr represent the probability that $verify(ret^*, \Gamma) = 1$. If Pr is a negligible value, we can conclude the outsourcing scheme is secure.*

3.2 Bilinear Pairings

Definition 2. *(Bilinear pairings): Having bilinear groups G_μ, G_ι, G_ζ with the prime order p. Then bilinear pairings $e : G_\mu \times G_\iota \to G_\zeta$ can be defined. It provides three properties:*

1) *Bilinearity: $e(m^x, n^y) = e(m, n)^{xy}$ with $x, y \in F_p^*$, $m \in G_\mu, n \in G_\iota$.*

2) *Non-degenercy: $e(m, n) \neq 1$.*

3) *Computation: For arbitrary $m \in G_\mu, n \in G_\iota$, $e(m, n)$ can be obtained easily.*

Definition 3. *(Co-CDH Assumption): Defining the bilinear groups G_μ, G_ι, G_ζ with the prime order p. Let $e : G_\mu \times G_\iota \to G_\zeta$. g and h are generators of G_μ and G_ι. Having $g^x \in G_\mu, h^y \in G_\iota$, $x, y \in F_p^*$, g^{xy} is difficult to be obtained with a negligible cost.*

4 Our Attacks

For the three schemes in [12] and [16], we only concentrete on the group-by sum, the similar attack can be used for the other two schemes. Here, firstly we review the scheme for group-by sum in [16]. Next, we show our first attack on [16] and prove the adversary can still forge tags. Finally, we give second attack on [16] and [12], we show cloud can use previous data to replace current data. Both of attacks are based on the scheme which can be queried many times, and the index sequence of data streams is same $i = 1, 2, 3...$

4.1 Review Wang's Scheme [16]

In their work, $ppara = (e, G_\mu, G_\zeta, p, g, g_1, g_2, g_3, \varphi_1, \varphi_2, \varphi)$. G_μ, G_ζ are bilinear groups with the prime order p. $e : G_\mu \times G_\mu \to G_\zeta$. $(g, g_1, g_2, g_3) \in G_\mu$ are generators. $\varphi, \varphi_1, \varphi_2$ are hash functions.

KeyGen (1^κ)**:** Using a secret number $b_j \in F_p^*$, D_j gets $pk_j = g^{b_j}$. Finally, it returns (b_j, pk_j).

GenTag $(\gamma_{j,i}, i)$**:** D_j computes tags $\rho_{j,i} = (g_1^{\varphi_1(D_j,i)} g_2^{\varphi_2(D_j,i)} g_3^{\gamma_{j,i}} \varphi(D_j, i))^{b_j}$. Consequently, it produces $(\rho_{j,i})$.

Compute $(\gamma_{j,i}, i, \rho_{j,i})$**:** For cloud, it computes $ret = \sum_{i\in\Lambda} \gamma_{j,i}$ and proof $\Gamma = \prod_{i\in\Lambda} \rho_{j,i}$. It produces (ret, Γ).

Verify (pk_j, ret, Γ)**:** For preprocessing, Clients obtain $\Psi_1 = \sum_{i\in\Lambda} \varphi_1(D_j, i), \Psi_2 = \sum_{i\in\Lambda} \varphi_2(D_j, i)$. Subsequently, they inspect

$$e(\Gamma, g) = e(g_1^{\Psi_1} g_2^{\Psi_2} g_3^{ret}, pk_j) \prod_{i\in\Lambda} e(\varphi(D_j, i), pk_j). \tag{1}$$

4.2 The First Attack on Wang's Scheme [16]

Here, we attack Wang's scheme and show that their scheme is still not secure. Specifically, let γ_j and γ'_j denote the data streams which are produced in first query and second query by the data source D_j respectively. Similarly, ρ_j and ρ'_j are tags in first query and second query. We also set $Z = g_3^{b_j}$ and it is a fixed value. Because tags ρ_j, ρ'_j and data streams γ_j, γ'_j are public, and hash computation can also be obtained, hence, the following equations can be gotten:

$$\rho_{j,i} = (g_1^{\varphi_1(D_j,i)} g_2^{\varphi_2(D_j,i)} g_3^{\gamma_{j,i}} \varphi(D_j, i))^{b_j} \tag{2}$$

$$\rho'_{j,i} = (g_1^{\varphi_1(D_j,i)} g_2^{\varphi_2(D_j,i)} g_3^{\gamma'_{j,i}} \varphi(D_j, i))^{b_j} \tag{3}$$

From Equations (2) and (3), we get

$$\begin{aligned}\frac{\rho_{j,i}}{\rho'_{j,i}} &= g_3^{b_j(\gamma_{j,i}-\gamma'_{j,i})} \\ &= Z^{(\gamma_{j,i}-\gamma'_{j,i})}\end{aligned}$$

Then we have $Z = (\frac{\rho_{j,i}}{\rho'_{j,i}})^{(\gamma_{j,i}-\gamma'_{j,i})^{-1}}$.

Let $N_i = (g_1^{\varphi_1(D_j,i)} g_2^{\varphi_2(D_j,i)} \varphi(D_j, i))^{b_j}$. Thus, their tags can also be described as

$$\begin{aligned}\rho_{j,i} &= (g_1^{\varphi_1(D_j,i)} g_2^{\varphi_2(D_j,i)} g_3^{\gamma_{j,i}} \varphi(D_j, i))^{b_j} \\ &= (g_1^{\varphi_1(D_j,i)} g_2^{\varphi_2(D_j,i)} \varphi(D_j, i))^{b_j} \cdot (g_3^{b_j})^{\gamma_{j,i}} \\ &= N_i Z^{\gamma_{j,i}}\end{aligned} \tag{4}$$

So we conclude $N_i = \frac{\rho_{j,i}}{Z^{\gamma_{j,i}}}$. Consequently, all N_i $(i \in [1, n])$ can be acquired by this way. In D_j, we find the value of N_i is fixed at i -th. Thus, if an adversary knows Z and N_i, it can forge tags for arbitrary data $\gamma_{j,i}$ by $\rho_{j,i} = N_i Z^{\gamma_{j,i}}$ distinctly.

4.3 Another Attack on Wang's Scheme and Liu's Scheme

In Wang's works [16], data source stores tags ρ_j and data stream γ_j on the cloud. Let γ'_j and ρ'_j denote the messages which are used previously. So when data source D_j sends the current data streams γ_j and the tags ρ_j to the cloud. As long as the index sequences are same, the cloud can use the previous messages γ'_j and ρ'_j to replace the current messages γ_j and ρ_j. The concrete attack is as the following steps.

Compute: Because the γ'_j and ρ'_j are stored on the cloud, when cloud receives the query for γ_j, it directly returns (ret', Γ') to clients, where $ret' = \sum_{i\in\Lambda} \gamma'_{j,i}$ and $\Gamma' = \prod_{i\in\Lambda} \rho'_{j,i}$.

Verify: Because clients only use the pk_j to check the correctness, obviously, the following equation can be passed the verification:

$$e(\Gamma', g) = e(g_1^{\Psi_1} g_2^{\Psi_2} g_3^{ret'}, pk_j) \prod_{i\in\Lambda} e(\varphi(D_j, i), pk_j), \tag{5}$$

where $\Psi_1 = \sum_{i\in\Lambda} \varphi_1(D_j, i), \Psi_2 = \sum_{i\in\Lambda} \varphi_2(D_j, i)$.

Intuitively, Liu's scheme can be attacked with a similar way [12]. This operation is easily done by the cloud and makes a wrong result $ret' = \sum_{i\in\Lambda} \gamma'_{j,i}$ to pass the verification.

5 The Proposed Schemes

Suppose there are three bilinear groups G_μ, G_ι, G_ζ with map $e : G_\mu \times G_\iota \to G_\zeta$ and the prime order p. $g_1, g_2 \in G_\mu$ and $h \in G_\iota$ are generators. $\varphi, \varphi_1, \varphi_2$ are hash functions and satisfy $(0, 1)^* \to Z_p^*$. δ is randomly chosen in F_p^*, it is secret. $\widetilde{h} = h^\delta$. Next it calculates $\varepsilon = \sum_{i=1}^n \varphi(\delta, i)$ and $\lambda_j = \sum_{i=1}^n \varphi(\delta, i) \cdot \varphi_2(D_j, i)$. Finally, $ppara = (e, G_\mu, G_\iota, G_\zeta, p, g_1, g_2, \widetilde{h}, \varphi, \varphi_1, \varphi_2, \varepsilon, \lambda_j)$. This step can be regarded as *preprocessing*.

Simultaneously, each data source must maintain a publicly synchronous clock t and it has a linear increment. When a new query arrives, it chooses the current clock

t. Therefore, this t is the same in each data source at any time. Then we use t to construct a evaluation key $GEK = g_1^{\varphi(\delta,t)}$ in group-by sum scheme (in inner product, it also need another evaluation key $IEK = h^{\varphi(\delta,t)}$). Since t is equal in each data source, hence, we have GEK, IEK are also equal. Moreover, even if a new data source must be added at some time, the new t just needs to synchronize with the previous t.

5.1 Group-By Sum

Here, we achieve the outsourcing of group-by sum. D_j sends the data streams γ_j to the cloud for the sum operation. We give our scheme as follows:

KeyGen $(1^\kappa, t)$**:** If a new query arrives, for all data sources, they choose the current clock t and compute evaluation key

$$GEK = g_1^{\varphi(\delta,t)}.$$

Then each data source arbitrarily setects a number b_j and computes public key $PK_j = h^{b_j}$. Finally, it outputs (b_j, PK_j, GEK).

GenTag $(\gamma_{j,i})$**:** Each data source computes the publicly verifiable tags

$$\rho_{j,i} = (g_1^{\varphi(\delta,t)(\varphi(\delta,i)+\varphi_1(D_j,i))} g_2^{\varphi_2(D_j,i)+\gamma_{j,i}})^{b_j}.$$

Finally, it outputs (ρ_j).

Compute (γ_j, ρ_j)**:** Cloud computes $ret = \sum_{i=1}^n \gamma_{j,i}$ and a proof

$$\Gamma = \prod_{i=1}^n \rho_{j,i}.$$

Finally, cloud returns (ret, Γ).

Verify $(\varepsilon, PK_j, GEK, ret, \Gamma)$ Firstly clients compute

$$\begin{aligned} \Psi_1 &= \sum_{i=1}^n \varphi_1(D_j, i), \\ \Psi_2 &= \sum_{i=1}^n \varphi_2(D_j, i). \end{aligned}$$

Then they check the following equation:

$$e(\Gamma, h) = e(GEK^{(\varepsilon+\Psi_1)} g_2^{\Psi_2+ret}, PK_j). \quad (6)$$

If the equation holds, clients output 1, otherwise, 0.

Correctness: Because of the correctness of the following equation, we say our scheme is correct.

$$\begin{aligned} e(\Gamma, h) &= e(\prod_{i=1}^n \rho_{j,i}, h) \\ &= e(\prod_{i=1}^n (g_1^{\varphi(\delta,t)(\varphi(\delta,i)+\varphi_1(D_j,i))} g_2^{\varphi_2(D_j,i)+\gamma_{j,i}})^{b_j}, h) \\ &= e(g_1^{\varphi(\delta,t)\sum_{i=1}^n (\varphi(\delta,i)+\varphi_1(D_j,i))} \\ &\quad \cdot g_2^{\sum_{i=1}^n \varphi_2(D_j,i)+\sum_{i=1}^n \gamma_{j,i}}, h^{b_j}) \\ &= e(g_1^{\varphi(\delta,t)(\varepsilon+\Psi_1)} g_2^{\Psi_2+ret}, PK_j) \\ &= e(GEK^{(\varepsilon+\Psi_1)} g_2^{\Psi_2+ret}, PK_j). \end{aligned}$$

5.2 Inner Product

Our scheme supports the query of inner product for any two data sources. For convenience, let data source D_1, D_2 send the data streams $\gamma_1 = \{\gamma_{1,1}, \gamma_{1,2}, ..., \gamma_{1,n}\}$ and $\gamma_2 = \{\gamma_{2,1}, \gamma_{2,2}, ..., \gamma_{2,n}\}$ to the cloud and the query is $ret_1 = \sum_{i=1}^n \gamma_{1,i} \cdot \gamma_{2,i}$. On the basis of group-by sum, D_j also needs to compute two publicly auxiliary keys AK_j, AK'_j and another evaluation key IEK. The specific inner product scheme has the following four phases.

KeyGen $(1^\kappa, t)$**:** If a new query arrives, for all data sources, they choose the current clock t and compute evaluation keys

$$GEK = g_1^{\varphi(\delta,t)}, IEK = h^{\varphi(\delta,t)}.$$

Then each data source arbitrarily setects a number b_j and computes public key $PK_j = h^{b_j}$. It also computes two auxiliary keys

$$\begin{aligned} AK_{j,i} &= (g_1^{\varphi(\delta,i)+\varphi_1(D_j,i)})^{b_j}, \\ AK'_{j,i} &= (g_1^{\varphi(\delta,i)+\varphi_1(D_j,i)})^{\delta b_j}. \end{aligned}$$

Finally, it outputs $(b_j, PK_j, GEK, IEK, AK_j, AK'_j)$.

GenTag $(\gamma_{j,i})$**:** Each data source computes the publicly verifiable tags

$$\rho_{j,i} = (g_1^{\varphi(\delta,t)(\varphi(\delta,i)+\varphi_1(D_j,i))} g_2^{\varphi_2(D_j,i)+\gamma_{j,i}})^{b_j}.$$

Finally, it outputs output (ρ_j).

Compute $(\rho_1, \rho_2, AK_1, AK'_1, \gamma_1, \gamma_2)$**:** Cloud computes

$$\begin{aligned} ret_1 &= \sum_{i=1}^n \gamma_{1,i}\gamma_{2,i}, \\ ret_2 &= \sum_{i=1}^n \varphi_2(D_1, i)\gamma_{2,i}, \\ \Gamma_1 &= \prod_{i=1}^n {\rho_{2,i}}^{\varphi_2(D_1,i)}, \\ \Gamma_2 &= \prod_{i=1}^n {AK_{1,i}}^{\gamma_{2,i}}, \\ \Gamma_3 &= \prod_{i=1}^n {AK'_{1,i}}^{\gamma_{2,i}}, \\ \Gamma_4 &= \prod_{i=1}^n {\rho_{1,i}}^{\gamma_{2,i}}. \end{aligned}$$

Finally, cloud returns $(ret_1, ret_2, \Gamma_1, \Gamma_2, \Gamma_3, \Gamma_4)$.

Verify $(\lambda_1, PK_1, PK_2, GEK, IEK, ret_1, ret_2, \Gamma_1, \Gamma_2, \Gamma_3, \Gamma_4)$: Firstly clients compute

$$\Psi_{12} = \sum_{i=1}^{n} \varphi_1(D_2, i)\varphi_2(D_1, i),$$
$$\Psi_{22} = \sum_{i=1}^{n} \varphi_2(D_2, i)\varphi_2(D_1, i).$$

Then they check the following equations:

$$e(\Gamma_1, h) = e(GEK^{\lambda_1+\Psi_{12}} g_2^{\Psi_{22}+ret_2}, PK_2) \quad (7)$$
$$e(\Gamma_3, h) = e(\Gamma_2, \widetilde{h}) \quad (8)$$
$$e(\Gamma_4, h) = e(\Gamma_2, IEK)e(g_2^{ret_2+ret_1}, PK_1) \quad (9)$$

If the above equations hold, clients accept and output 1, otherwise, 0.

Discussion: In our scheme, intuitively, the result of $ret_1 = \sum_{i=1}^{n} \gamma_{1,i} \cdot \gamma_{2,i}$ is only related to $\Gamma_4 = \prod_{i=1}^{n} \rho_{1,i}{}^{\gamma_{2,i}}$, the detailed structure of proof Γ_4 is $\Gamma_4 =$

$$(g_1^{\varphi(\delta,t)\sum_{i=1}^{n}(\varphi(\delta,i)+\varphi_1(D_1,i))\gamma_{2,i}} g_2^{\sum_{i=1}^{n}\varphi_2(D_1,i)\gamma_{2,i}+ret_1})^{b_1}.$$

Accordingly, the clients can not check ret_1 until they know the values of $\sum_{i=1}^{n}(\varphi(\delta, i) + \varphi_1(D_1, i))\gamma_{2,i}$ and $ret_2 = \sum_{i=1}^{n} \varphi_2(D_1, i)\gamma_{2,i}$. Thus, we also utilize cloud to compute $\Gamma_2 = (g_1^{\sum_{i=1}^{n} (\varphi(\delta,i)+\varphi_1(D_1,i))\gamma_{2,i}})^{b_1}$ and ret_2. Respectively, the proof Γ_3 and Γ_1 can check the correctness of Γ_2 and ret_2.

Correctness: For Equation (7):

$$\begin{aligned}
e(\Gamma_1, h) &= e(\prod_{i=1}^{n} \rho_{2,i}{}^{\varphi_2(D_1,i)}, h) \\
&= e((\prod_{i=1}^{n} g_1^{\varphi(\delta,t)(\varphi(\delta,i)+\varphi_1(D_2,i))} g_2^{\varphi_2(D_2,i)+\gamma_{2,i}})^{b_2\varphi_2(D_1,i)}, h) \\
&= e(g_1^{\varphi(\delta,t)(\sum_{i=1}^{n}\varphi(\delta,i)\varphi_2(D_1,i)+\sum_{i=1}^{n}\varphi_1(D_2,i)\varphi_2(D_1,i))} \\
&\quad \cdot g_2^{\sum_{i=1}^{n}\varphi_2(D_2,i)\varphi_2(D_1,i)+\sum_{i=1}^{n}\gamma_{2,i}\varphi_2(D_1,i)}, h^{b_2}) \\
&= e(g_1^{\varphi(\delta,t)(\lambda_1+\Psi_{12})} g_2^{\Psi_{22}+ret_2}, PK_2) \\
&= e(GEK^{\lambda_1+\Psi_{12}} g_2^{\Psi_{22}+ret_2}, PK_2).
\end{aligned}$$

For Equation (8): First, we have:

$$\Gamma_2 = \prod_{i=1}^{n} AK_{1,i}{}^{\gamma_{2,i}} = \prod_{i=1}^{n} (g_1^{\varphi(\delta,i)+\varphi_1(D_1,i)})^{b_1\gamma_{2,i}}$$

Then

$$\begin{aligned}
e(\Gamma_3, h) &= e(\prod_{i=1}^{n} AK'_{1,i}{}^{\gamma_{2,i}}, h) \\
&= e(\prod_{i=1}^{n} (g_1^{\varphi(\delta,i)+\varphi_1(D_1,i)})^{\delta b_1\gamma_{2,i}}, h) \\
&= e(\prod_{i=1}^{n} (g_1^{\varphi(\delta,i)+\varphi_1(D_1,i)})^{b_1\gamma_{2,i}}, h^{\delta}) \\
&= e(\Gamma_2, \widetilde{h})
\end{aligned}$$

For Equation (9): we have

$$\begin{aligned}
e(\Gamma_4, h) &= e(\prod_{i=1}^{n} \rho_{1,i}^{\gamma_{2,i}}, h) \\
&= e((\prod_{i=1}^{n} g_1^{\varphi(\delta,t)(\varphi(\delta,i)+\varphi_1(D_1,i))} g_2^{\varphi_2(D_1,i)+\gamma_{1,i}})^{b_1\gamma_{2,i}}, h) \\
&= e(\prod_{i=1}^{n} g_1^{\varphi(\delta,t)(\varphi(\delta,i)+\varphi_1(D_1,i))\gamma_{2,i}} \\
&\quad \cdot \prod_{i=1}^{n} g_2^{(\varphi_2(D_1,i)+\gamma_{1,i})\gamma_{2,i}}, h^{b_1}) \\
&= e(\prod_{i=1}^{n} g_1^{\varphi(\delta,t)(\varphi(\delta,i)+\varphi_1(D_1,i))\gamma_{2,i}}, h^{b_1}) \\
&\quad \cdot e(g_2^{\sum_{i=1}^{n}\varphi_2(D_1,i)\gamma_{2,i}+\sum_{i=1}^{n}\gamma_{1,i}\gamma_{2,i}}, h^{b_1}) \\
&= e(\prod_{i=1}^{n} g_1^{(\varphi(\delta,i)+\varphi_1(D_1,i))b_1\gamma_{2,i}}, h^{\varphi(\delta,t)}) e(g_2^{ret_2+ret_1}, PK_1) \\
&= e(\Gamma_2, IEK)e(g_2^{ret_2+ret_1}, PK_1).
\end{aligned}$$

Discussion: Only considering single scheme, the original $ppara$ can be divided into $ppara1$ and $ppara2$ in our two schemes. For Group-by sum, $ppara1 = (e, G_\mu, G_\iota, G_\zeta, p, g_1, g_2, \varphi, \varphi_1, \varphi_2, \varepsilon)$, and for inner product, $ppara2 = (e, G_\mu, G_\iota, G_\zeta, p, g_1, g_2, \widetilde{h}, \varphi, \varphi_1, \varphi_2, \lambda_j)$. The size of $ppara1$ is constant, but the size of $ppara2$ is $l|Z_p|$. Hence, when l is large, it is impossible for clients to store them. The technique to handle this problem is that the clients can use a trusted third party to store these public parameters.

6 Security Analysis

Firstly we prove the tags can not be forged and cloud can not replace input data. Subsequently, following our experiment, the security was proved.

Theorem 1. *Under our first attack, the correct tags can not be forged by any adversary.*

Because the public messages that belongs to the group-by sum are a part of the inner product, therefore, we mainly focus on the inner product.

Proof. Suppose clients request the data source D_j to produce data streams twice for inner product query. In first

query, we set $t=1$, then we have $GEK = g_1^{\varphi(\delta,1)}, IEK = h^{\varphi(\delta,1)}$, and the date streams and tags are γ_j and ρ_j. In second query, we set $t=2$, so $GEK = g_1^{\varphi(\delta,2)}, IEK = h^{\varphi(\delta,2)}$. Meanwhile, γ'_j denotes data streams and ρ'_j denotes tags. With setting $Y = g_2^{b_j}$, we have

$$\rho_{j,i} = (g_1^{\varphi(\delta,1)(\varphi(\delta,i)+\varphi_1(D_j,i))} g_2^{\varphi_2(D_j,i)+\gamma_{j,i}})^{b_j},$$

$$\rho'_{j,i} = (g_1^{\varphi(\delta,2)(\varphi(\delta,i)+\varphi_1(D_j,i))} g_2^{\varphi_2(D_j,i)+\gamma'_{j,i}})^{b_j}.$$

Next we divide $\rho_{j,i}$ with $\rho'_{j,i}$.

$$\begin{aligned}\frac{\rho_{j,i}}{\rho'_{j,i}} =& (g_1^{b_j})^{(\varphi(\delta,1)-\varphi(\delta,2))(\varphi(\delta,i)+\varphi_1(D_j,i))} (g_2^{b_j})^{\gamma_{j,i}-\gamma'_{j,i}} \\ =& AK_{j,i}{}^{\varphi(\delta,1)-\varphi(\delta,2)} Y^{\gamma_{j,i}-\gamma'_{j,i}}\end{aligned}$$

Intuitively, $AK_{j,i}$ is not fixed and $\varphi(\delta,1)-\varphi(\delta,2)$ is secret, so it is impossible to get the value of Y.

In the mean time, if evaluation key $GEK = g_1^{\varphi(\delta,1)}$ and $GEK' = g_1^{\varphi(\delta,2)}$ are used to reconstruct tags, we have

$$\frac{GEK}{GEK'} = g_1^{\varphi(\delta,1)-\varphi(\delta,2)}.$$

Since GEK is fixed in each query, so we set $F = \left(\frac{GEK}{GEK'}\right)^{b_j}$. Then

$$\begin{aligned}\frac{\rho_{j,i}}{\rho'_{j,i}} =& \left(\frac{GEK}{GEK'}\right)^{b_j(\varphi(\delta,i)+\varphi_1(D_j,i))} Y^{\gamma_{j,i}-\gamma'_{j,i}} \\ =& F^{\varphi(\delta,i)+\varphi_1(D_j,i)} Y^{\gamma_{j,i}-\gamma'_{j,i}}.\end{aligned}$$

Obviously, since $\varphi(\delta,i)$ is secret and not fixed, we can not find a fixed value to denote $F^{\varphi(\delta,i)+\varphi_1(D_j,i)}$, thus the value of Y still can not be obtained. Therefore, the tags can not be forged in our schemes. □

Theorem 2. *Under our second attack, the malicious cloud cannot replace the input data.*

Here, we only discuss the proof of group-by sum, and the proof of inner product is similar to group-by sum.

Proof. In our scheme, the tags are

$$\rho_{j,i} = (g_1^{\varphi(\delta,t)(\varphi(\delta,i)+\varphi_1(D_j,i))} g_2^{\varphi_2(D_j,i)+\gamma_{j,i}})^{b_j}.$$

Intuitively, due to the update of $\varphi(\delta,t)$, the same data $\gamma_{j,i}$ have different tags $\rho_{j,i}$ at the different times. For example, when clients have group-by sum query twice for D_j, we set $t=1$ or 2. Therefore, we have:

$$\rho_{j,i} = (g_1^{\varphi(\delta,1)(\varphi(\delta,i)+\varphi_1(D_j,i))} g_2^{\varphi_2(D_j,i)+\gamma_{j,i}})^{b_j},$$

$$\rho'_{j,i} = (g_1^{\varphi(\delta,2)(\varphi(\delta,i)+\varphi_1(D_j,i))} g_2^{\varphi_2(D_j,i)+\gamma'_{j,i}})^{b_j}.$$

Naturally, clients must use $GEK = g_1^{\varphi(\delta,1)}$ to verify $\Gamma = \prod_{i=1}^n \rho_{j,i}$ and use $GEK' = g_1^{\varphi(\delta,2)}$ to verify $\Gamma' = \prod_{i=1}^n \rho'_{j,i}$. If the cloud uses $\gamma'_{j,i}$ to replace $\gamma_{j,i}$, our *verify* algorithm will output 0. Moreover, from *Theorem 1*, we know the adversary can not forge tags for arbitrary data stream. Therefore, it can not use previous input data to replace current input data. □

Theorem 3. *For group-by sum, under the co-CDH assumption, our scheme is secure.*

Proof. We suppose there has an adversary A who can win $Exp_A^{1^k}$, then we can use challenger B takes advantage of A to break co-CDH assumption. Concretely, knowing the values of $g_2^a, h^{b_j^*}$, B can compute $g_2^{ab_j^*}$.

Step 1: Challenger B randomly selects five numbers $a, b_j^*, c, \delta^*, T \in F_p^*$. Then it sets $g'_1 = g_2'^{c^{-1}}$, $g'_2 = g_2^a$, $\hat{h} = h^{\delta^*}$ and randomly generates two vector $(Z_i)_n, (w_{j,i})_n \in F_p^*$, $(1 \le i \le n)$. It computes $\varphi(\delta,i) = cZ_i$ and sets $GEK = g_1'^T$. Next it calculates $\varepsilon^* = \sum_{i=1}^n \varphi(\delta,i)$. Finally B returns $PK'_j = h^{b_j^*}$ and GEK to A.

Step 2: Adaptively, adversary A requests the challenger B to generate public information on this data streams γ_j. Specifically, when challenger B received the inputs, it computes $\varphi_1(D_j,i) = -cZ_i + cb_j^{*-1} w_{j,i}$, $\varphi_2(D_j,i) = -b_j^{*-1} w_{j,i}\delta^* - \gamma_{j,i}$ and returns to A. At last, B computes tags

$$\rho^*_{j,i} = (g_1'^{T(\varphi(\delta,i)+\varphi_1(D_j,i))} g_2'^{\varphi_2(D_j,i)+\gamma_{j,i}})^{b_j^*} = g_2^{aw_{j,i}(T-\delta^*)}$$

and returns them to A.

Step 3: Challenger B requests A to calculate $ret = \sum_{i=1}^n \gamma_{j,i}$. According to challenge input (D_j, γ_j), adversary A gives a result (ret^*, Γ) with $ret^* \neq \sum_{i=1}^n \gamma_{j,i}$.

Step 4: If the wrong result (ret^*, Γ) can be verified, the following equation can be obtained.

$$\begin{aligned}\Gamma =& (GEK^{(\varepsilon^*+\Psi_1)} g_2'^{\Psi_2+ret^*})^{b_j^*} \\ =& (g_1'^{T(\varepsilon^*+\Psi_1)} g_2'^{\Psi_2+ret^*})^{b_j^*} \\ =& (g_1'^{T\sum_{i=1}^n(\varphi(\delta,i)+\varphi_1(D_j,i))} g_2'^{\sum_{i=1}^n \varphi_2(D_j,i)+ret^*})^{b_j^*} \\ =& (g_2^{ac^{-1}T\sum_{i=1}^n cZ_i - cZ_i + cb_j^{*-1} w_{j,i}} \\ & \cdot g_2^{a(\sum_{i=1}^n(-b_j^{*-1}w_{j,i}\delta^*-\gamma_{j,i}))+a\cdot ret^*})^{b_j^*} \\ =& g_2^{aT\sum_{i=1}^n w_{j,i}} g_2^{-a\delta^*\sum_{i=1}^n w_{j,i} - ab_j^*\sum_{i=1}^n \gamma_{j,i} + ab_j^*\cdot ret^*} \\ =& g_2^{aT\sum_{i=1}^n w_{j,i} - a\delta^*\sum_{i=1}^n w_{j,i}} g_2^{ab_j^*(\sum_{i=1}^n -\gamma_{j,i}+ret^*)} \\ =& g_2^{a(T-\delta^*)\sum_{i=1}^n w_{j,i}} g_2^{ab_j^*(ret^*-ret)}\end{aligned}$$

if $ret^* - ret \neq 0$, we have:

$$g_2^{ab_j^*} = \left(\frac{\Gamma}{g_2^{a(T-\delta^*)\sum_{i=1}^n w_{j,i}}}\right)^{(ret^*-ret)^{-1}}$$

□

Theorem 4. *If Γ_1 can pass the verification, we say ret_2 is correct.*

Proof.

Step 1: Similarly with the proof of *Theorem 1*, challenger B randomly selects six numbers $a, b_1^*, b_2^*, c, \delta^*, T \in F_p^*$. Then it sets $g_1' = g_2'^{c^{-1}}, g_2' = g_2^a, \hat{h} = h^{\delta^*}$ and randomly generates three vector $(Z_i)_n, (w_{1,i})_n, (w_{2,i})_n \in F_p^*$, $(1 \le i \le n)$. It computes $\varphi(\delta, i) = cZ_i$. Let $GEK = g_1'^T$, $IEK = h^T$. For data sources D_1,D_2, challenger B computes the public keys $PK_1' = h^{b_1^*}, PK_2' = h^{b_2^*}$ and returns them.

Step 2: Adaptively, adversary A requests the challenger B to generate public information on this inputs γ_1, γ_2. Specifically, when challenger B received the inputs, it computes $\varphi_1(D_j, i) = -cZ_i + cb_j^{*-1} w_{j,i}$, $\varphi_2(D_j, i) = -b_j^{*-1} w_{j,i}\delta^* - \gamma_{j,i}$, where $j = 1$ or 2, and sends them to A. Next, it achieves $\lambda_1^* = \sum_{i=1}^n \varphi(\delta, i)\varphi_2(D_1, i)$. At last, B calculates

$$\rho_{1,i}^* = g_2^{aw_{1,i}(T-\delta^*)},$$
$$\rho_{2,i}^* = g_2^{aw_{2,i}(T-\delta^*)}$$

and gives them to A.

Step 3: Challenger B requests A to compute $ret_2 = \sum_{i=1}^n \varphi_2(D_1, i)\gamma_{2,i}$. According to challenge tuple $(D_1, \gamma_1), (D_2, \gamma_2)$, adversary A shows a result (ret_2^*, Γ_1) with $ret_2^* \neq \sum_{i=1}^n \varphi_2(D_1, i)\gamma_{2,i}$.

Step 4: If the wrong result (ret_2^*, Γ_1) can be verified, the following equation can be obtained.

$$\begin{aligned}
\Gamma_1 =&(GEK^{\lambda_1^*+\Psi_{12}} g_2'^{\Psi_{22}+ret_2^*})^{b_2^*} \\
=&(g_1'^{T(\lambda_1^*+\Psi_{12})} g_2'^{\Psi_{22}+ret_2^*})^{b_2^*} \\
=&(g_2^{ac^{-1}T\sum_{i=1}^n \varphi(\delta,i)\varphi_2(D_1,i)+\varphi_1(D_2,i)\varphi_2(D_1,i)} \\
&\cdot g_2^{a\sum_{i=1}^n \varphi_2(D_2,i)\varphi_2(D_1,i)+a\cdot ret_2^*})^{b_2^*} \\
=&(g_2^{ac^{-1}T\sum_{i=1}^n \varphi_2(D_1,i)(\varphi(\delta,i)+\varphi_1(D_2,i))} \\
&\cdot g_2^{a\sum_{i=1}^n(-b_2^{*-1}w_{2,i}\delta^*-\gamma_{2,i})\varphi_2(D_1,i)+a\cdot ret_2^*})^{b_2^*} \\
=&g_2^{ac^{-1}b_2^*T\sum_{i=1}^n \varphi_2(D_1,i)(cZ_i-cZ_i+cb_2^{*-1}w_{2,i})} \\
&\cdot g_2^{a\delta^*\sum_{i=1}^n -w_{2,i}\varphi_2(D_1,i)-ab_2^*\sum_{i=1}^n \gamma_{2,i}\varphi_2(D_1,i)+ab_2^*\cdot ret_2^*} \\
=&g_2^{aT\sum_{i=1}^n w_{2,i}\varphi_2(D_1,i)} g_2^{a\delta^*\sum_{i=1}^n -w_{2,i}\varphi_2(D_1,i)+ab_2^*(ret_2^*-ret_2)} \\
=&g_2^{aT\sum_{i=1}^n w_{2,i}\varphi_2(D_1,i)-a\delta^*\sum_{i=1}^n w_{2,i}\varphi_2(D_1,i)} g_2^{ab_2^*(ret_2^*-ret_2)} \\
=&g_2^{a(T-\delta^*)\sum_{i=1}^n w_{2,i}\varphi_2(D_1,i)} g_2^{ab_2^*(ret_2^*-ret_2)}
\end{aligned}$$

where

$$\Psi_{12} = \sum_{i=1}^n \varphi_1(D_2, i)\varphi_2(D_1, i)$$
$$\Psi_{22} = \sum_{i=1}^n \varphi_2(D_2, i)\varphi_2(D_1, i)$$

if $ret_2^* - ret_2 \neq 0$, we have:

$$g_2^{ab_2^*} = \left(\frac{\Gamma_1}{g_2^{a(T-\delta^*)\sum_{i=1}^n w_{2,i}\varphi_2(D_1,i)}}\right)^{(ret_2^*-ret_2)^{-1}}$$

□

Theorem 5. *If Γ_3 can pass the verification,we say Γ_2 is correct.*

Proof. We assume that there has a false Γ_2^* makes the Γ_3 pass the verification. Here, Γ_2 is correct and $\Gamma_2^* \neq \Gamma_2$. We deduce

$$\begin{aligned}
\Gamma_3 =&(\Gamma_2^*)^\delta \\
=&\left(\frac{\Gamma_2^*}{\Gamma_2}\cdot\Gamma_2\right)^\delta \\
=&\left(\frac{\Gamma_2^*}{\Gamma_2}\right)^\delta \cdot \Gamma_3
\end{aligned}$$

Because $\delta \neq 0$, so the above equation dose not hold. Thus, when equation holds, Only $\Gamma_2 = \Gamma_2^*$. □

Theorem 6. *As long as the cloud computes Γ_4 honestly, we can say ret_1 is correct.*

Proof.

Step 1: Expect for the following steps, the security proof for inner product is similar to the *Theorem4*.

Challenger B computes two auxiliary keys AK_j, AK_j' and uploads to A. We have $j = 1$ or 2, then

$$\begin{aligned}
AK_{j,i} =&(g_1'^{\varphi(\delta,i)+\varphi_1(D_j,i)})^{b_j^*} = g_2^{aw_{j,i}}, \\
AK_{j,i}' =&g_1'^{(\varphi(\delta,i)+\varphi_1(D_j,i))\delta^* b_j^*} = g_2^{a\delta^* w_{j,i}}.
\end{aligned}$$

Step 2: this step is same as *Theorem 4*.

Step 3: Challenger B requests A to calculate $ret_1 = \sum_{i=1}^n \gamma_{1,i}\gamma_{2,i}$, then adversary A shows $(ret_1^*, ret_2, \Gamma_1, \Gamma_2.\Gamma_3, \Gamma_4)$ with $ret_1^* \neq \sum_{i=1}^n \gamma_{1,i}\gamma_{2,i}$.

Step 4: If the wrong result $(ret_1^*, ret_2, \Gamma_1, \Gamma_2.\Gamma_3, \Gamma_4)$ can be verified, the following equation can be obtained.

$$\begin{aligned}
\Gamma_4 =& \Gamma_2^T (g_2'^{ret_2+ret_1^*})^{b_1^*} \\
=&\prod_{i=1}^n AK_{1,i}^{\gamma_{2,i}T} (g_2'^{\sum_{i=1}^n \varphi_2(D_1,i)\gamma_{2,i}+ret_1^*})^{b_1^*} \\
=&g_1'^{T\sum_{i=1}^n(\varphi(\delta,i)+\varphi_1(D_1,i))b_1^*\gamma_{2,i}} \\
&\cdot (g_2'^{\sum_{i=1}^n(-b_1^{*-1}w_{1,i}\delta^*-\gamma_{1,i})\gamma_{2,i}+ret_1^*})^{b_1^*} \\
=&g_2^{aT\sum_{i=1}^n w_{1,i}\gamma_{2,i}} \\
&\cdot g_2^{a\delta^*\sum_{i=1}^n -w_{1,i}\gamma_{2,i}-ab_1^*\sum_{i=1}^n \gamma_{1,i}\gamma_{2,i}+ab_1^*\cdot ret_1^*} \\
=&g_2^{aT\sum_{i=1}^n w_{1,i}\gamma_{2,i}-a\delta^*\sum_{i=1}^n w_{1,i}\gamma_{2,i}} g_2^{ab_1^*(ret_1^*-ret_1)} \\
=&g_2^{a(T-\delta^*)\sum_{i=1}^n w_{1,i}\gamma_{2,i}} g_2^{ab_1^*(ret_1^*-ret_1)}
\end{aligned}$$

if $ret_1^* - ret_1 \neq 0$,we have:

$$g_2^{ab_1^*} = \left(\frac{\Gamma_4}{g_2^{a(T-\delta^*)\sum_{i=1}^n w_{1,i}\gamma_{2,i}}}\right)^{(ret_1^*-ret_1)^{-1}}$$

Theorem 7. *The scheme for inner product is secure.*

Proof. Obviously, the security of inner product is related to *Theorem 4, Theorem 5, Theorem 6*. If *Theorem 4, Theorem 5, Theorem 6* can be proved, then the security of inner product can also be proved. □

7 Performance Evaluation

We give the evaluation from three aspects: Communication, storage, and computation. In group-by sum, we mainly compare with Wang's scheme [16] and Liu's scheme [12]. In inner product, we mainly compare with Liu's scheme [12] scheme and Zhang's scheme [21]. And we conduct the transfer of public parameters on the offline period.

7.1 Communication Cost

We mainly assess the communication cost of two stages: *data source→cloud* and *cloud→clients*. For the transfer of public parameters and evaluation keys, the communication cost is $l|Z_p|$.

In group-by sum:

Data source → cloud: All Liu's [12], Wang's [16] and our scheme are $n|G| + n|Z_p|$.

Cloud → clients: All three schemes are constant.

In inner product:

Data source → cloud: In Liu's [12], the communication cost is $n|G| + n|Z_p|$. Intuitively, in Zhang's [21] and our scheme, the communication cost is $3n|G| + n|Z_p|$. But it can be divided into online cost and offline cost, and two schemes have a same cost in offline phase and online phase, $2n|G|$ and $n|G| + n|Z_p|$.

Cloud → clients: All three schemes are constant.

Consequently, the communication cost is same on these three schemes.

7.2 Storage Cost

In this part, we mainly consider the storage cost in data source and clients. Simultaneously, we store the tags, auxiliary keys and data streams on the cloud.

In group-by sum: The data source only stores the private/public keys (b_j, pk_j, GEK, t). Consquently, at side of data source and clients, all three schemes are constant.

In inner product: Similar to group-by sum, all three schemes are constant, except for the cost of our scheme is $l|Z_p|$ at side of clients.

Intuitively, compared with previous schemes, our storage cost is higher.

7.3 Computation Cost

As cloud have the huge computation power, consequently, we mainly consider the computation cost in data source and clients. Then we use $O_{Z/m/e/p}$ to denote the time complexity of operations. O_Z is multiplication in Z_p. O_m is the multiplication in G_μ, G_ι, G_ζ. O_e is the exponentiation in G_μ, G_ι, G_ζ. O_p is the bilinear pairing. In *KeyGen and Preprocessing*, because public key PK_j, evaluation key IEK, GEK, auxiliary keys AK, AK', and system parameters *ppara* can be precomputed in the offline phase, consequently we do not discuss the cost of this phase. The cost of each phase is shown in Table 1 and Table 2.

In group-by sum: In *GenTag*, the cost of Liu's [12] is $O_e(3 + 3n) + O_m(2n)$. Then Wang *et al.* [16] use an modified *GenTag* algorithm and the cost is $O_e(3 + 4n) + O_m(3n)$. But in our scheme, the cost is $O_e(2 + 2n) + O_m(n) + O_Z(1)$. In *verify*, Liu's [12], Wang's [16] and our scheme have a same offline cost. Only considering online computation, both our scheme and Liu's [12] are constant. Specifically, the cost of Liu's [12] is $O_p(2) + O_e(3) + O_m(2)$, the cost of our scheme is $O_p(2) + O_e(2) + O_m(1)$. But in Wang's [16], their cost is $O_p(3) + O_e(3) + O_m(n + 2)$. Therefore, our scheme is efficient.

In Inner product: Similar to previous analysis, in *GenTag*, Liu's [12] is $O_e(3 + 3n) + O_m(2n)$, our cost is $O_e(2 + 2n) + O_m(n) + O_Z(1)$. Zhang's [21] can be divided into offline cost and online cost, $O_e(6+4n) + O_m(2n + 1) + O_Z(1)$, $O_e(2 + 2n) + O_m(n) + O_Z(1)$. In *verify*, Both Liu's [12] and our scheme have precomputation, and the cost is $O_Z(4n)$ and $O_Z(2n)$ separately. For online stage, Liu's, Zhang's and our scheme have a constant cost, $O_p(6) + O_e(9) + O_m(6)$, $O_p(4) + O_e(2) + O_m(2)$, $O_p(7) + O_e(3) + O_m(2)$. With the comparison, our scheme is more efficient than Liu's [12].

8 Experiment Evaluation

In this section, we give our experiments from two aspects: the number of data sources and the size of data streams. We use Miracl Library and Intel Core i7 CPU, 1.8 GHz, 4GB RAM. Fairly, we use the tate pairing under type-1 pairings.

Firstly, we concentrate on the experiment evaluation when the number of data sources l is in the range of 2 to 32. In the mean time, we assume that any data source D_j only performs the operation of inner product with D_1 and the dimension of data streams is 2000. Thus, as revealed in Figure 2 and Figure 3, the time cost have a linear relationship with the number of data sources. Noting that the main cost is at the stage of *GenTag* and *Compute*, we get that clients can process data with a low overhead.

Then, we consider the impact of the dimension of data streams on our scheme. Here, we show our experiments

Table 1: Comparison of computation cost for Group -by sum

	KeyGen and Preprocessing($ppara1$)	GenTag	Compute	Verify
Liu's [12]	$O_e(1)$	$O_e(3+3n)+O_m(2n)$	$O_m(n-1)$	$O_p(2)+O_e(3)+O_m(2)$
Wang's [16]	$O_e(1)$	$O_e(3+4n)+O_m(3n)$	$O_m(n-1)$	$O_p(3)+O_e(3)+O_m(n+2)$
Our	$O_e(2)$	$O_e(2+2n)+O_m(n)+O_Z(1)$	$O_m(n-1)$	$O_p(2)+O_e(2)+O_m(1)$

Table 2: Comparison of computation cost for inner product

	KeyGen and Preprocessing($ppara2$)	GenTag	Compute	Verify
Liu's [12]	$O_e(1)$	$O_e(3+3n)+O_m(2n)$	$O_Z(3n)+O_e(3n)$ $+O_m(3(n-1))$	$O_Z(4n)+O_p(6)$ $+O_e(9)+O_m(6)$
Zhang's [21]	$O_e(3)$	$O_e(8+6n)+$ $O_m(3n+1)+O_Z(2)$	$O_Z(n)+O_e(3n)$ $+O_m(3(n-1))$	$O_p(4)+O_e(2)$ $+O_m(2)$
Our	$O_e(2n+6)+O_Z(1+ln)$	$O_e(2+2n)+$ $O_m(n)+O_Z(1)$	$O_Z(2n)+O_e(4n)$ $+O_m(4(n-1))$	$O_Z(2n)+O_p(7)$ $+O_e(3)+O_m(2)$

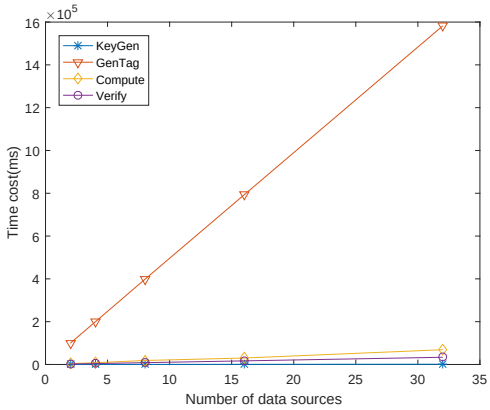


Figure 2: Each stage on group-by sum

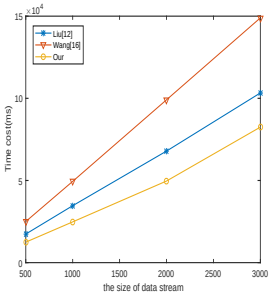


Figure 4: Group tag

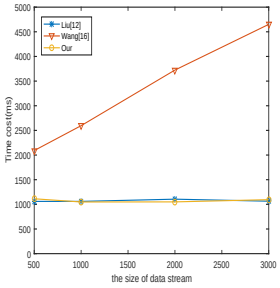


Figure 5: Group verify

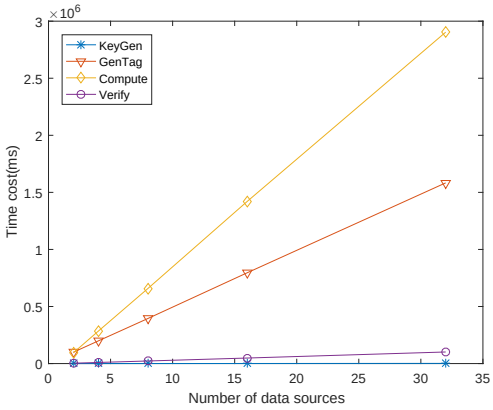


Figure 3: Each stage on inner product

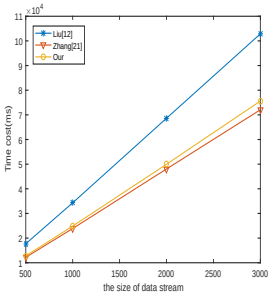


Figure 6: Inner tag

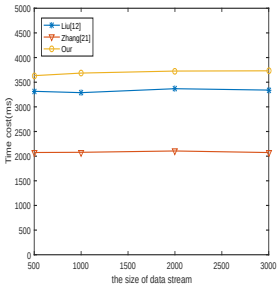


Figure 7: Inner verify

with Liu's [12], Wang's [16] and Zhang's [21] and just compare the online cost with all messages have been preprocessed. The Figure 4 and Figure 5 are the experiment comparisons of group-by sum in *GenTag* and *Verify*. Similarly, the Figure 6 and Figure 7 are the experiment comparisons of inner product. From Figure 4, Figure 5 and Figure 6, we get our schemes are more efficient. Meanwhile, although our scheme has a larger cost in Figure 7, it is constant.

9 Conclusion

In this paper, firstly we find there still are some security flaws in the previous works. Then we give two secure schemes for group-by sum and inner product under multi-keys. For security, our schemes not only have no security flaws, but also are secure under the co-CDH assumption. Experiment result proves that our schemes are efficient. Althought our schemes have a preprocessing cost in the side of communication and storage, comparing with our improved aspects, our schemes still have many practical applications. In future, we will extend our schemes to the matrix multiplication and solve this problem.

Acknowledgments

This work was supported by the National Natural Science Foundation of China under Grants (No. 61902314, 62072371, 61772418), the Natural Science Basic Research Plan in Shaanxi Province of China under Grants No.2018JZ6001, basic Research Program of Qinghai Province under Grants No. 2020-ZJ-701.

References

[1] M. Backes, D. Fiore, and R. M. Reischuk, "Verifiable delegation of computation on outsourced data," in *Proceedings of ACM SIGSAC Conference on Computer & Communications Security*, pp. 863–874, 2013.

[2] S. G. Choi, J. Katz, R. Kumaresan, and C. Cid, "Multi-client non-interactive verifiable computation," in *Theory of Cryptography Conference*, pp. 499–518, 2013.

[3] K. Elkhiyaoui, M. Önen, M. Azraoui, and R. Molva, "Efficient techniques for publicly verifiable delegation of computation," in *Proceedings of the 11th ACM on Asia Conference on Computer and Communications Security*, pp. 119–128, 2016.

[4] D. Fiore and R. Gennaro, "Publicly verifiable delegation of large polynomials and matrix computations, with applications," in *Proceedings of ACM Conference on Computer and Communications Security*, pp. 501–512, 2012.

[5] D. M. Freeman, "Converting pairing-based cryptosystems from composite-order groups to prime-order groups," in *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 44–61, 2010.

[6] R. Gennaro, C. Gentry, and B. Parno, "Non-interactive verifiable computing: Outsourcing computation to untrusted workers," in *Annual Cryptology Conference*, pp. 465–482, 2010.

[7] D. Gil, A. Ferrández, H. Mora-Mora, and J. Peral, "Internet of things: A review of surveys based on context aware intelligent services," *Sensors*, vol. 16, no. 7, pp. 1069, 2016.

[8] X. Lei, X. Liao, T. Huang, and F. Heriniaina, "Achieving security, robust cheating resistance, and high-efficiency for outsourcing large matrix multiplication computation to a malicious cloud," *Information Sciences*, vol. 280, pp. 205–217, 2014.

[9] X. Lei, X. Liao, T. Huang, H. Li, and C. Hu, "Outsourcing large matrix inversion computation to a public cloud," *IEEE Transactions on Cloud Computing*, vol. 1, no. 1, pp. 1–1, 2013.

[10] T. Li, C. Gao, L. Jiang, W. Pedrycz, and J. Shen, "Publicly verifiable privacy-preserving aggregation and its application in IoT," *Journal of Network and Computer Applications*, vol. 126, pp. 39–44, 2019.

[11] L. Liu and Y. Liu, "A note on one outsourcing scheme for large-scale convex separable programming," *International Journal of Electronics and Information Engineering*, vol. 12, no. 4, pp. 155–161, 2020.

[12] X. Liu, W. Sun, H. Quan, W. Lou, Y. Zhang, and H. Li, "Publicly verifiable inner product evaluation over outsourced data streams under multiple keys," *IEEE Transactions on Services Computing*, vol. 10, no. 5, pp. 826–838, 2016.

[13] S. Nath and R. Venkatesan, "Publicly verifiable grouped aggregation queries on outsourced data streams," in *IEEE 29th International Conference on Data Engineering (ICDE'13)*, pp. 517–528, 2013.

[14] S. Papadopoulos, G. Cormode, A. Deligiannakis, and M. Garofalakis, "Lightweight authentication of linear algebraic queries on data streams," in *Proceedings of ACM SIGMOD International Conference on Management of Data*, pp. 881–892, 2013.

[15] Q. Su, J. Yu, C. Tian, H. Zhang, and R. Hao, "How to securely outsource the inversion modulo a large composite number," *Journal of Systems and Software*, vol. 129, pp. 26–34, 2017.

[16] X. A. Wang, Y. Liu, A. K. Sangaiah, and J. Zhang, "Improved publicly verifiable group sum evaluation over outsourced data streams in IoT setting," *Computing*, vol. 101, no. 7, pp. 773–790, 2019.

[17] J. Yang, Y. Li, and Y. Ren, "Novel and secure outsourcing algorithms for multiple bilinear pairings with single untrusted server," *International Journal Network Security*, vol. 21, no. 5, pp. 872–880, 2019.

[18] J. Ye, Z. Xu, and Y. Ding, "Secure outsourcing of modular exponentiations in cloud and cluster computing," *Cluster Computing*, vol. 19, no. 2, pp. 811–820, 2016.

[19] X. Yu, Z. Yan, and A. V. Vasilakos, "A survey of verifiable computation," *Mobile Networks and Applications*, vol. 22, no. 3, pp. 438–453, 2017.

[20] S. Zhang, C. Tian, H. Zhang, J. Yu, and F. Li, "Practical and secure outsourcing algorithms of matrix operations based on a novel matrix encryption method," *IEEE Access*, vol. 7, pp. 53823–53838, 2019.

[21] S. Zhang, H. Li, Y. Dai, M. He, and R. Lu, "EPP-DMM: An efficient and privacy-protected delegation scheme for matrix multiplication," in *IEEE Global Communications Conference*, pp. 1–6, 2017.

[22] S. Zhang, H. Li, Y. Dai, J. Li, M. He, and R. Lu, "Verifiable outsourcing computation for matrix multiplication with improved efficiency and applicability," *IEEE Internet of Things Journal*, vol. 5, no. 6, pp. 5076–5088, 2018.

[23] X. Zhang, T. Jiang, K. C. Li, A. Castiglione, and X. Chen, "New publicly verifiable computation for batch matrix multiplication," *Information Sciences*, vol. 479, pp. 664–678, 2019.

[24] J. Zheng, H. Yang, Y. Su, and J. Qin, "A blindly public verifiable outsourcing scheme for matrix multiplication," *Journal of Shandong Univerty (Natural Science)*, vol. 54, no. 11, pp. 1–11, 2019.

[25] L. Zhou, Y. Zhu, and K. K. R. Choo, "Efficiently and securely harnessing cloud to solve linear regression and other matrix operations," *Future Generation Computer Systems*, vol. 81, pp. 404–413, 2018.

Biography

Zhiqiang Du received his B.S. degree from Xi'an Shiyou University, Xi'an, China, in 2018. He now is a M.S. degree candidate in Electronic and Communication Engineering with the School of Xi'an University of Posts and Telecommunications, Xi'an, China. His research interests include secure outsourcing computation.

Dong Zheng received Ph.D degree from Xidian University in 1999. He then joined the School of Information Security Engineering, Shanghai JiaoTong University. He is currently a professor in Xi'an University of Post and Telecommunications, China. His research interests include information theory, cryptography and information security. He is a senior member of Chinese Association for Cryptologic Research and member of Chinese Communication Society.

Qinglan Zhao received her Ph.D. degree (2018) Shanghai Jiao Tong University, Shanghai, China. Since 2014, she has been an associate professor in Xi'an University of Post and Telecommunications, Xi'an, China. Her research interests focus on cryptographic functions and symmetric cryptography.

A Pseudorandom Bit Generator Based on Gaussian Function

Qi Wu

(Corresponding author: Qi Wu)

Department of Electronic Commerce, Jiangxi University of Finance & Economics
Shuanggang Street, Nanchang 330013, China
Email: wuqiocjzd@126.com

(Received Nov. 30, 2020; Revised and Accepted May 6, 2021; First Online Nov. 10, 2021)

Abstract

This article proposes for the first time rules which new 1-Dimensional Discrete Chaotic Mapping should follow in research on chaotic cryptography. Via translation and scale to the independent and dependent variables of Gaussian Function, a novel 1-Dimensional Discrete Chaotic Mapping is obtained. Bifurcation Diagram and Lyapunov Exponent Spectrum indicate that the mapping owns a broad chaotic area and is suitable for chaotic cryptography. Next, based on the mapping, a pseudorandom bit generator is designed. Pseudorandom tests illustrate that the mapping is the 2nd best one so far. Therefore, it possesses great application prospects.

Keywords: 1-Dimensional Discrete Chaotic Mapping; Gaussian Function; Pseudorandom Bit Generator

1 Introduction

Of all the discrete chaotic systems [3–5], 1-Dimensional Discrete Chaotic Mappings (abbreviated as 1DDCM) are the simplest and of the highest efficiency [3]. Our experiments over these years [1, 6, 8, 12–22, 24, 25] have, however, proved that two classic 1DDCMs, *i.e.* Logistic mapping and piecewise linear mapping (skew tent mapping for most), to which most attention is paid, are flawed. The chaotic area of Logistic mapping is extremely narrow and incontinuous, which incurs severe difficulty in selecting strong parameters. Although the chaotic area of skew tent mapping is full, when applied to construction of Pseudorandom bit Generator (abbreviated as PRBG), its strong cipher space is constrained in a small adjacent area of 0.5 [2, 9, 10, 23].

It could not be denied that Logistic mapping and skew mapping contribute a lot to 1DDCM. Logistic mapping is the fastest chaotic mapping (Logistic mapping is quadratic, whose degree couldn't be lowered, as linear mapping could never be a chaotic mapping.) As far as we know, skew tent mapping is the only mapping with full chaotic area. Most chaotic mappings possess both periodic area and chaotic area. Nevertheless, for many application fields, such as information security, parameters could only be drawn from chaotic area. Thus, skew tent mapping provides the largest scope for these fields. However, these contributions couldn't offer a solid base for cryptographic applications. Their tiny strong cipher space facilitates brute force attacks launched by adversaries.

As said by Ref. [3], unimodal mappings often own chaotic properties. Nonetheless, Ref. [3] doesn't elaborate on which kind of unimodal mappings could own chaotic properties. After many failures and several successes [14,15,21,22], we obtain some immature knowledge for designing new 1DDCM, as shown below:

1) The mapping must be a surjection.
   This is the main reason why Logistic mapping is feasible only when u equals 4 or comes extremely near 4. Non-surjection degrades dramatically during iterations.

2) The mapping must be axisymmetric, or, at least approaches axisymmetry as closely as possible.
   This is the main reason why the strong cipher space of skew tent mapping is confined in an extremely small adjacent area of 0.5. This is also the main reason why Ref. [14, 22] is the best 1DDCM so far.

3) The mapping shouldn't contain too many segments.
   Nowadays, some Ref. [7, 11] try to design new 1DDCM via making it own many segments, which is often called as "sawtooth mapping". I deem this way infeasible, as skew tent mapping is dramatically slower than Logistic mapping, due to the time-consuming branch structure. Employing too many segments, although the cipher space could be enlarged easily, it lowers the efficiency heavily. I think it's totally impractical.

4) The mapping shouldn't go too far away from polynomials.

In the course "mathematical statistics", a few functions are exponential, whose exponent contains integral. These functions could be cleared from our concern, as their efficiency must be rather low. Polynomials are of the highest efficiency.

In a word, informally speaking, to design an excellent unimodal 1DDCM, we should make it go through the point (0.5,1), and try to make it as axisymmetric on the line x=0.5 as possible, and try to avoid many segments. I then take Gaussian function into account, and obtain results only worse than Ref. [22]. As far as I know, it's the 2nd best 1DDCM all over the world, and I hope that it'll shed some light on researchers of chaotic cryptology.

The upcoming parts are as follows: Section 2 introduces a chaotic mapping based on a variant of Gaussian function. Section 3 attempts to apply it to the construction of PRBG. Section 4 concludes.

2 A Chaotic Mapping Based on a Variant of Gaussian Function

It's known that, Gaussian function is:

$$y = ae^{-(x-b)^2/(2c^2)}. \tag{1}$$

Obviously, Equation (1) has already contained some concern on translation & scale of independent variable & dependent variable. Among elementary transformations, rotation is one of the commonest. However, it will introduce $\sin(\cdot)$ & $\cos(\cdot)$ (Both are implemented by Taylor expansion, which is rather time-consuming.). Therefore, this paper discards rotation. We only employ translation and scale to obtain a variant of Gaussian function.

For this step, I discard my brain, ignoring the translation & scale already contained in Equation (1), put on the translation & scale parts for independent variable & dependent variable to acquire a new function:

$$y = \frac{d_2 a}{e^{\frac{(d_1 x + f_1 - b)^2}{2c^2}}} - f_2. \tag{2}$$

In Equation (2), f_1 and f_2 serve for the translation of independent variable and dependent variable respectively, d_1 and d_2 serve for the scale of independent variable and dependent variable respectively.

First, demanding Equation (2) go through points (0, 0) and (1, 0), we have:

$$\begin{cases} \frac{d_2 a}{e^{\frac{(f_1-b)^2}{2c^2}}} - f_2 = 0 \\ \\ \frac{d_2 a}{e^{\frac{(d_1+f_1-b)^2}{2c^2}}} - f_2 = 0 \end{cases} \tag{3}$$

Let $y' = -d_2 a e^{-\frac{(d_1 x + f_1 - b)^2}{(2c^2}} \cdot \frac{d_1}{c^2}(d_1 x + f_1 - b) = 0$, we have

$$x^* = \frac{b - f_1}{d_1} \tag{4}$$

Let Equation (2) go through point $(x^*, 1)$, we have

$$d_2 a - f_2 = 1. \tag{5}$$

Substituting Equation (5) into Equation (3) and Equation (3), we have

$$\begin{cases} \frac{1+f_2}{e^{\frac{(f_1-b)^2}{2c^2}}} = f_2 \\ \\ \frac{1+f_2}{e^{\frac{(d_1+f_1-b)^2}{2c^2}}} = f_2 \end{cases} \tag{6}$$

From Equation (6), we have

$$f_2[e^{\frac{(f_1-b)^2}{2c^2}} - 1] = 1. \tag{7}$$

From Equation (6), we have

$$f_2[e^{\frac{(d_1+f_1-b)^2}{2c^2}} - 1] = 1. \tag{8}$$

From Equation (7) and Equation (8), via a little elementary calculation, we have

$$d_1 = 2(b - f_1). \tag{9}$$

From Equation (9), look back at Equation (4), we know $x^* = 0.5$, which exactly fits the requirement in Section 1. Hence, we could adjust the form of y as follows:

$$\begin{aligned} y &= \frac{1 + f_2}{e^{\frac{(d_1 x - 0.5 d_1)^2}{2c^2}}} - f_2 \\ &= \frac{1 + f_2}{e^{\frac{d_1^2 (x-0.5)^2}{2c^2} r}} - f_2. \end{aligned} \tag{10}$$

Next, replace the parameters in Equation (10). (Notice that, the letters after replacement have nothing to do with the previous ones!) Let $f_2 = a$, $\frac{d_1^2}{2c^2} = b$, we have

$$y = \frac{1 + a}{e^{b(x-0.5)^2}} - a. \tag{11}$$

It's easy to find that, Equation (11) automatically satisfies going through point (0.5,1), whereas the 2 requirements, *i.e.* going through points (0, 0) and (1, 0), become one equation:

$$1 + a = ae^{\frac{b}{4}}. \tag{12}$$

i.e.

$$\frac{1}{a} + 1 = e^{\frac{b}{4}}. \tag{13}$$

Putting $\ln(\cdot)$ to both ends, we have

$$4 \ln(\frac{1}{a} + 1) = b. \tag{14}$$

From the basic knowledge of logarithm, we know it's required that $\frac{1}{a} + 1 > 0$. From the image of inverse proportional function, we know it's required that $a < -1$ or

$a > 0$. Once the value of a is fixed, the value of b is settled by Equation (14) automatically.

In conclusion, the variant of Gaussian function Equation (11) acquired in this paper owns only one free parameter a. Once a is fixed in $(-\infty, -1) \cup (0, +\infty)$, b is settled accordingly. Thus, the entire mapping Equation (11) is fixed.

For convenience hereafter, we name the new mapping Equation (11) Gaussian Function's Variant Chaotic Mapping (often abbreviated as GFVCM).

Next, let's analyze its chaotic properties.

For GFVCM, set $x_0 = 0.1$, let a go from -100 to -1.1 and 0.1 to 100 with Step 0.1. For the 1990 parameters, iterate the system 500 times respectively, filtering the first 200 times, draw the x value for the last 300 times as shown in Figure 1.

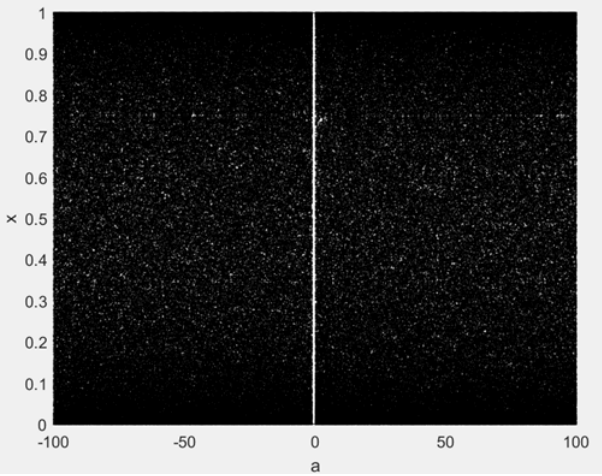


Figure 1: Bifurcation diagram

From Figure 1 it could be seen that, for the aforementioned initial values and parameters, GFVCM doesn't have any obvious periodic area and is quite suitable for PRBG. The white vertical line in Figure 1 is because a skips interval [-1, 0].

Set $x_0 = 0.1$, let a go from -100 to -1.1 and 0.1 to 100 with Step 0.1. For the 1990 parameters, iterate the system 2000 times, filtering the first 1000 times, calculate the Lyapunov exponent from the last 1000 times as shown in Figure 2.

From Figure 2 it could be seen that, for the initial values and parameters mentioned above, the Lyapunov exponent of GFVCM is positive for the most part, *i.e.* the system always dwells in chaotic area. Hence, it fits PRBG perfectly.

3 A PRBG Based on GFVCM

This paper designs PRBG as the same as Ref. [22] did. Given x_0, a, GFVCM obtains a new xi after each iteration, compares it with 0.5 to emit a new bit:

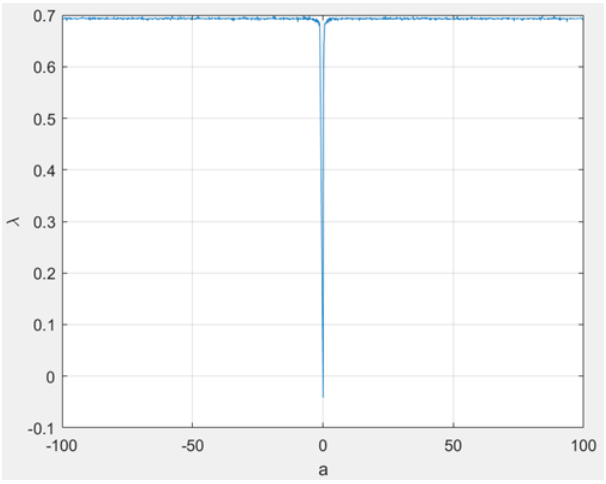


Figure 2: Lyapunov exponent spectrum

$$s_i = \begin{cases} 0 & x_i < 0.5 \\ 1 & x_i \geq 0.5 \end{cases} \tag{15}$$

In Ref. [22], when c goes from -1000 to 0 with Step 0.01, for the 100001 parameters, there're 60841 ones passing all the 5 pseudorandom tests, *i.e.*, about 60% parameters are strong, which is the champion of 1DDCM. As to the PRBG in this paper, when a goes from -100 to -1.001 and 0.001 to 100 with Step 0.001, for the 199000 parameters, this result becomes 94612, *i.e.*, approximately 48% parameters are strong. Although it's worse than Ref. [22], it's already the silver medalist of 1DDCM, overwhelming Logistic mapping and skew tent mapping.

Next, for $x_0 = 0.1$, this paper tests 3 bit streams of length 50000 with a set to -57, -16, 99 respectively and acquires results under significance level 0.05. This paper omits all the basic knowledge for tests. Reader interested in them could refer to Ref. [12–22, 24].

Tables 1 - 5 illustrate that, all the 3 bit streams have passed the 5 pseudorandom tests. As BM algorithm is too time-consuming, this paper sets the length of bit steams to 1000 while computing Table 6 with all the other conditions unchanged. It's obvious that all the 3 bit streams own excellent Linear Complexity (All is close to BSS.).

Table 1: Results of monobit test

a	χ^2	Critical Value
-57	0.2785	3.84
-16	1.2103	
99	0.0353	

Table 2: Results of serial test

a	χ^2	Critical Value
-57	0.3851	5.99
-16	1.2781	
99	3.1592	

Table 3: Results of poker test

a	$\chi^2(m=4)$	Critical Value
-57	19.5443	25
-16	10.3821	
99	17.8445	

Table 4: Results of runs test

a	χ^2	Critical Value
-57	16.8623	31.4
-16	27.4338	
99	18.6306	

Table 5: Results of auto-correlation test

a	$\|\chi\|(d=10000)$	Critical Value
-57	0.98	1.96
-16	0.26	
99	1.52	

Table 6: Results of linear complexity

a	Linear Complexity	N/2
-57	501	500
-16	500	
99	499	

4 Conclusion

Based on Gaussian Function, after translation and scale, we obtain a variant mapping with 1 free parameter. Both Bifurcation diagram and Lyapunov Exponent Spectrum demonstrate that the new mapping possesses wonderful chaotic properties. Based on it, a PRBG is devised. Its strong cipher space is smaller compared to our best result, but it's already overwhelmed Logistic mapping and skew tent mapping. All the statistical tests illustrate that, the bit streams generated own excellent pseudo randomness and superior linear complexity. In the future, we decide to test other 1DDCM and try to excel our best result.

Acknowledgments

This study is funded by the Science and Technology Project of Provincial Education Department of Jiangxi for Youth (GJJ180288). Thanks to my uncle Shiqian Wu for his valuable advice and suggestions.

References

[1] J. Bi, S. Yin, H. Li, L. Teng, C. Zhao, "Research on medical image encryption method based on improved Krill Herb algorithm and chaotic systems", *International Journal of Network Security*, vol. 22, no. 3, pp. 486-491, 2020.

[2] K. Charif, A. Drissi, Z. El A. Guenno.n, "A pseudo random number generator based on chaotic billiards," *International Journal of Network Security*, vol. 19, no. 3, pp. 479-486, 2017.

[3] B. L. Hao, *Starting with Parabolas*, Shanghai Science and Technology Education Press, Shanghai, 2003.

[4] Y. B. Huang, S. H. Wang, Y. Wang, Y. Zhang, Q. Y. Zhang, "A hyperchaotic encrypted speech perceptual hashing retrieval algorithm based on 2D-Gabor transform", *International Journal of Network Security*, vol. 23, no. 5, pp. 924-935, 2021.

[5] C. Li, G. Luo, C. Li, "An image encryption scheme based on the three-dimensional chaotic logistic map", *International Journal of Network Security*, vol. 21, no. 1, pp. 22-29, 2019.

[6] D. Liu, F. Wang, H. Wang, "A BP neural network-oriented Henon hyperchaotic system for image encryption", *International Journal of Network Security*, vol. 23, no. 4, pp. 698-705, 2021.

[7] S. J. Luo, S. S. Qiu, X. Chen, "Spatiotemporal chaotic one-way hash function construction based on coupled tent maps," *Journal of Shenzhen University Science and Engineering*, vol. 29, no.4, pp. 335-340, 2012.

[8] D. S. A. Minaam, M. A. Ibrahim, E. Badr, "Chaotic NHCP: Building an efficient secure framework for cloud computing environment based on chaos theory", *International Journal of Network Security*, vol. 22, no. 2, pp. 283-295, 2020.

[9] C. Nouar, Z. E. Guennoun, "A pseudo random bit generator based on a modified chaotic map", *International Journal of Network Security*, vol. 21, no. 3, pp. 402-408, 2019.
[10] N. K. Pareek, V. Patidar, K. K. Sud, "A random bit generator using chaotic maps", *International Journal of Network Security*, vol. 10, no. 1, pp. 32-38, 2010.
[11] X. Su, G. Yang, "An encryption system of virtual optical lens imaging based on coupled sawtooth chaos map," *Computer Technology and Development*, vol. 24, no. 1, pp. 167-171, 2014.
[12] Z. W. Tan, Q. Wu, "Study of linearly cross-coupled chaotic systems for a random bit generator," in *Proceedings International Conference on Computational Intelligence and Security*, 2008.
[13] Z. W. Tan, Q. Wu, "Study of exponentially cross-coupled chaotic systems for a random bit generator," in *Proceedings International Symposium on Intelligent Information Technology Application*, pp. 224-227, 2008.
[14] Q. Wu, "A chaos-based hash function," in *Proceedings International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery*, pp. 1-4, 2015.
[15] Q. Wu, "A quartic chaotic mapping," in *Proceedings International Conference on Computer Science and Information Security*, 2016.
[16] Q. Wu, "An independent variable exclusively coupled chaotic system for a pseudorandom bit generator," in *Proceedings International Conference on Industrial Informatics – Computing Technology, Intelligent Technology, Industrial Information Integration*, pp. 341-344, 2016.
[17] Q. Wu, "Independent variable exclusively coupled chaotic pseudorandom bit generator," *Computer Engineering & Science*, vol. 38, no. 11, pp. 2197-2201, 2016.
[18] Q. Wu, "A dependent variable harmonically coupled chaotic system for a pseudorandom bit generator," in *Proceedings International Conference on Smart Materials, Intelligent Manufacturing and Automation*, 2018.
[19] Q. Wu, "A dependent variable exclusively coupled chaotic system for a pseudorandom bit generator," in *Proceedings International Conference on Network and Information Systems for Computers*, 2018.
[20] Q. Wu, "A pseudorandom bit generator based on a dependent variable exclusively coupled chaotic system," in *Proceedings International Conference on Intelligent Information Processing*, pp. 11-16, 2018.
[21] Q. Wu, "Another look at a proposed quartic chaotic mapping," in *Proceedings International Symposium on Advances in Electrical, Electronics and Computer Engineering*, 2019.
[22] Q. Wu, "Another look at a proposed cubic chaotic mapping," in *Proceedings International Symposium on Cyberspace Safety and Security*, pp. 407-412, 2019.
[23] Q. Wu, "An independent variable swinging coupled chaotic system for a pseudorandom bit generator", *International Journal of Network Security*, vol. 21, no. 5, pp. 883-887, 2021.
[24] Q. Wu, Z. W. Tan, C. X. Wan, "A harmonically coupled chaotic system for a pseudo-random bit generator," *Journal of Chinese Computer System*, vol. 32, no. 4, pp. 639-643, 2011.
[25] C. Xu, "A novel chaotic image encryption algorithm based on bit-level permutation and extended ZigZag transform", *International Journal of Network Security*, vol. 23, no. 2, pp. 255-260, 2021.

Biography

Qi Wu, born in 1984, is a Phd & lecturer of Jiangxi University of Finance & Economics. His research field is information security & chaos theory. He has published more than twenty papers and has finished several projects.

Secure and Accountable Data Access Control Against Malicious Behavior in Smart Grids

Leyou Zhang[1], Chao Song[2], and Yi Mu[3]
(Corresponding author: Leyou Zhang & Chao Song)
School of Mathematics and Statistics, and school of Cyber Engineering, Xidian University[1]
School of Cyber Engineering, Xidian University[2]
Xi'an, Shanxi 710071, China
Email: lyzhang@mail.xidian.edu.cn, songchao_xidian@163.com
College of Mathematics and Informatics, Fujian Normal University[3]
(Received Nov. 4, 2020; Revised and Accepted May 6, 2021; First Online Nov. 13, 2021)

Abstract

The development of the Internet of Things (IoT) has helped the smart grid address many of the security challenges and weaknesses of the traditional grid. Due to the special characteristics of the smart grid transmission data, malicious entity (users or authority) can obtain private information, such as electricity habits, by analyzing the users' power consumption characteristics. The existing accountable schemes focus on connectivity, but not for security purposes. Securely sharing sensitive data with fine-grained access control to them is a serious problem in smart grids. In this paper, we aim to find a global method based on encryption techniques to cover the security requirements and privacy protection in sharing smart grid data. We also propose a black-box accountable scheme based on attribute-based encryption (ABE). This scheme uses the ABE technology to achieve fine-grained access control. And on this basis, it uses token-based and protocol interaction to achieve black-box traceability of users and accountability of authorities. Security analysis and performance comparisons show that our scheme is more effective than previous schemes.

Keywords: Attribute-based Encryption; Authority Accountable; Black-Box Traceable; Smart Grid

1 Introduction

The smart grid can not only use diversified renewable energy as a source of electrical energy but also realize automatic and intelligent management of the grid. It is characterized by self-healing, reliability, interaction, economy, efficiency, compatibility, and resource optimization. In particular, its two-way transmission of electrical energy means that users can input excess storage power into the power transmission network according to market and grid requirements [11]. As shown in Figure 1, the distributed smart grid has a hierarchical structure for data acquisition and power transmission control. Power transmission consists of power generation, transmission, distribution and power consumption, and includes a main control center (MCC), multiple district control centers (DCCs) and multiple substations. A substation contains remote terminal units (RTUs), circuit breakers, communication equipment, and routing network management. Each substation covers a Neighborhood Area Network (NAN), and each NAN contains one or more base stations and multiple building area networks (BANs). The NAN gateway can monitor how much power is transmitted to a specific area. Each BAN contains a specific number of home area networks (HANs), and either WiFi or ZigBee is generally used for communication between a BAN and a HAN. In this way, the smart grid is divided into several hierarchical networks, which was introduced in [3].

Smart meters are in charge of collecting power consumption information and monitoring data from smart devices through an HAN, and sends information to the MCC through the BAN and the NAN [14]. In this process, the protection of sensitive data and the authentication of legal users have become problems faced by smart grids. At the same time, different users want to access power information according to different needs, so as to ensure the safety and privacy of the grid [25].

There are many methods based on cryptographic techniques to assure the confidentiality, integrity, and accountability of the data shared in the smart grid, but most of them are based on a one-to-one mechanism and only ensure the security from data. In 2005, attribute-based encryption (ABE) for one-to-many mechanism was proposed [28]. The ABE was derived from identity-based encryption (IBE), which provides fine-grained access control [7]. In the smart grid, a user's power consumption information, environmental monitoring information (humidity), system load monitoring information (voltage), equipment fault information (power outage) and other

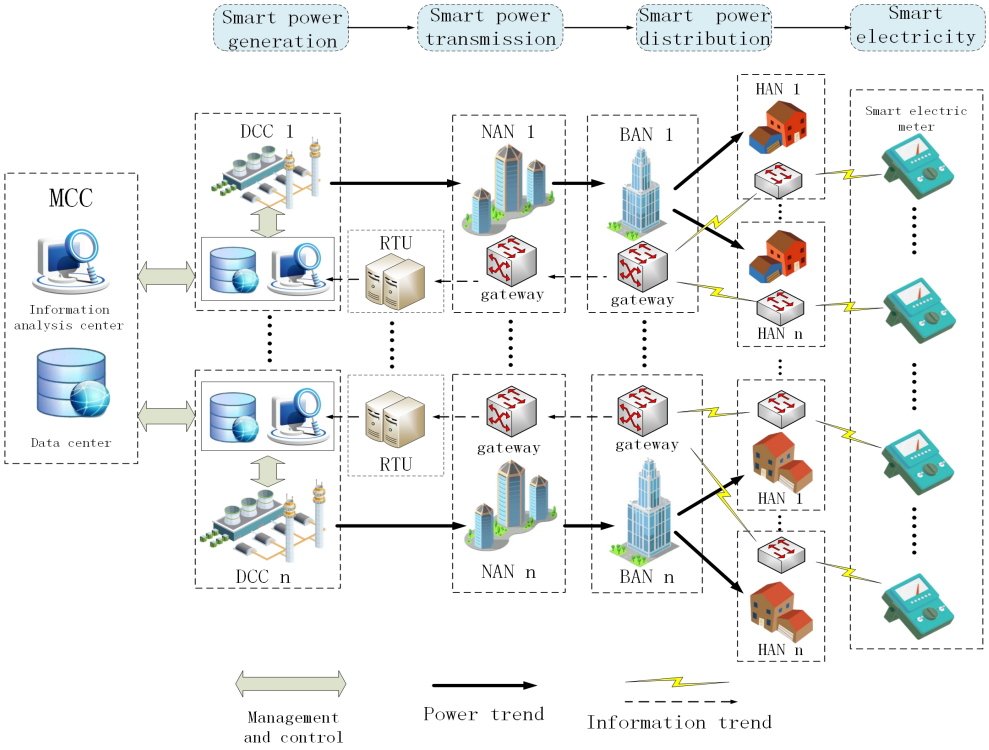


Figure 1: Smart grid structure

information can be employed as user attributes, which is an effective method for implementing access control. Ciphertext-policy attribute-based encryption (CP-ABE) is used to eliminate security threats in the smart grid. In CP-ABE, attribute sets are contained in secret user keys, and the access policy is contained in ciphertext. Only attribute sets that satisfy access policy can decrypt data file.

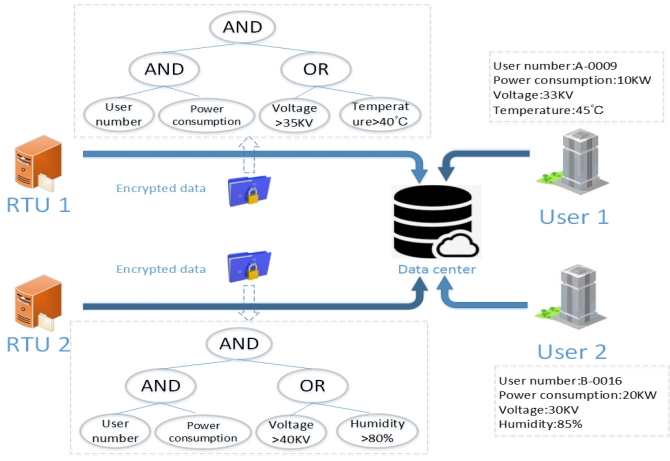


Figure 2: Example of smart grid

For example, suppose the attribute set collected by one user is {user number: A-0009, power consumption: 10 kW, voltage: 33 kV, temperature: 45°C }, and the attribute set collected by another user is {user number: B-0016, power consumption: 20 kW, voltage: 30 kV, humidity: 85% }. DCC1 and DCC2, respectively, feed back two commands: "User temperature of the first area is over 40°C degrees power failure", "User humidity of second area is greater than 80% power failure", and the access strategy is W1={(user number) AND (power consumption) AND (voltage>35 kV) OR (temperature > 40 °C)}, W2 = {(user number) AND (power consumption) AND (voltage > 40 kV) OR (humidity > 80%)}, so only the first user can execute the command of first area, and the second user can execute the command of second area. This is shown in Figure 2.

1.1 Motivation

Many traditional ABE schemes still face several problems. In reality, user and authority are not trustworthy. For one thing, a malicious user (which is authorized) could use a secret key to generate decryption devices, then provide the decryption device to other unauthorized users and profit from it. For another, if a secret key is fully controlled by authority, the authority can then generate secret keys, and if authority uses the secret keys to engage in malicious activities, it will not be captured and prosecuted. Moreover, there have been few studies on user traceability and authority accountability in the smart grid.

Traceability. Under normal circumstances, the use of the grid requires network access. The user provides identity information and related attributes, then the secret key is distributed by the data management de-

partment in the control center (CC). The key allows the user to view the power information on the internet. As an example, consider the situation in which a mall needs to enter the network as a user, and then provides the attribute information of all the shops to obtain the private key. The mall can now view the electricity usage information of all the shops from the network, assuming that the secret key is utilized to generate a decryption device by the shopping mall's electricity management department. Furthermore, unauthorized users could now obtain a shop's electricity usage data of the store through the decryption device and profit from it. This is why there is a need to set up a traceable mechanism to trace the departments that leak information.

Accountable Authority. In the smart grid, if the secret key distribution is completely controlled by the authority, then it could engage in malicious activities that will lead to the leakage of information and invasion of privacy. In addition, it is hard to determine whether the authority has leaked the private keys or not. To this end, it is necessary to provide an authoritative and accountable mechanism to determine whether it is the private key of the authority leak.

1.2 Related Works

As part of a smart city, the smart grid has been studied by researchers. Bose proposed the smart grid structure in [3] and summarized the application of smart grid and the supporting infrastructure. Kim *et al.* divided the smart grid infrastructure into two types [9]. One is the power infrastructure, which includes related electrical equipment such as transformers and electricity meters, and the information infrastructure, which is mainly used for auxiliary equipment for communication, while ensuring safe operation of the power grid and reliable power transportation. In the smart grid, security is a basic requirement. Liang *et al.* proposed the installation of fiber-optic cable, a mobile link, and data monitoring and acquisition equipment to ensure the security of the grid [14], as well as ethical behavior on the part of users and service providers. To prevent military threats, the National Institute of Standards and Technology (NIST) has insisted on requirements of security in smart grids [11].

To securely store data collected by smart meters, fine-grained access control of data is required. Fadlullah *et al.* and Bobba *et al.* proposed fine-grained access control schemes for the smart grid [2, 4]. Ruj *et al.* proposed a framework [26], which includes two schemes, one for data aggregation between three regional networks, and one for using ABE to ensure data collection security of RTUs and user access control. However, ABE cannot prevent malicious users from abusing private key permissions. For solving this problem, Li *et al.* first proposed an accountable ABE scheme [12, 13] to support users and authorities center accountability. After that, Liu *et al.* proposed two CP-ABE schemes for tracing malicious users' accountability. One is a white-box tracing feature with weak concepts [16], which was first proposed by Goyal [5]; The other was a black-box tracing feature with strong concepts [15], which was introduced in [6]. As an extension, Liu *et al.* further proposed ABE schemes [17] and [18] with a sub-linear length ciphertext. Later, Yu *et al.* proposed an accountable CP-ABE scheme with multi-authority [32]. Zhao *et al.* proposed a black-box traceable ABE scheme with multi-authority [37]. And Zhang *et al.* proposed a white-box traceable CP-ABE scheme with multi-authority [34]. In their schemes, a trace algorithm can trace the malicious entity which leaked credentials.

Since ABE inherits the method of key escrow in IBE, a corrupt key distribution authority has the ability to fully generate private keys. In order to weaken the central authority, Goyol *et al.* proposed the concept of accountability [6]. Wang *et al.* proposed a KP-ABE scheme [30] that included accountability mechanisms and traceability features, which is a collusion-resistant traceability. After that, Ning *et al.* and Zhang *et al.* came up with two CP-ABE schemes for accountable authority [21] and [35], respectively, which are white-box traceable. Later, Zhang *et al.* applied the traceable technology to smart health to prevent malicious users from revealing private information [36]. Ning *et al.* proposed two kinds of traceable schemes for malicious users in the cloud environment [20]. Yu *et al.* proposed accountable CP-ABE scheme in could [31]. And Liu *et al.* proposed traceable and revocable CP-ABE scheme [19]. Their schemes are constructed on the composite order group with white-box traceability and do not support authority accountability. Jiang *et al.* came up with a CP-ABE scheme about the problem of secret key abuse in fog computing [8], Wang *et al.* proposed a CP-ABE scheme for cloud storage [29]. These schemes are white-box traceable, but cannot support authority accountability. Qiao *et al.* proposed a black-box traceable scheme [24] which is based on [1]. Its access structure is a tree structure, but it does not support the authority accountability mechanism. Lai *et al.* proposed a general and accountable ABE construction method [10], and this framework can implement large universe and the tracing of malicious entities. However, it has no specific implementation scheme. In addition, the efficiency of the trace algorithm needs to be applied to the actual ABE scheme, and its efficiency needs to be tested and verified.

1.3 Our Contributions

We extend the [38] to the ABE scheme and strengthen its selective security to adaptive security. We present a black-box accountable data access control mechanism and the main contributions of this paper are as follows.

Black-box traceability. Compared with white-box traceability, which is only trace the owner of secret key, black-box traceability can be traced back to the creator of the decoding box generated using the

secret key. We use two tokens t_k and t_c to mark the secret key and ciphertext. Then it can symmetrically decrypt the extracted identity information for malicious user. Once the normal user decrypts, it can only be decrypted correctly when $t_k \neq t_c$. The trace algorithm feature can use this token to achieve the purpose of black-box traceability.

Accountable authority. For dishonest authorities that use the secret key to generate a decrypted black box, the trace algorithm can also be used to track the authority. The tracing algorithm can also trace a result for a good form of private key. We use a protocol instead of a key generation algorithm to weaken the right of authority. Authority interacts with the user through a secret key generation protocol. This scheme has a token mechanism to control the private key, that is, the authority completes part of token and generates part of private key. The user then completes the construction of the entire private key. User and authority use zero-knowledge proof interaction to prove that user-generated components are valid.

Stronger security. Compared with the selective security model, which declare a challenge before the game, adaptive security is more stronger. In the game, the adversary will declare public parameters after publishing the challenge ciphertext in the selective security model. We verify later that the scheme is adaptive security.

1.4 Organization

The rest of this paper is divided into the following sections. Section 2 presents preliminaries. Section 3 defines the system model and security model. Section 4 presents the concrete construction process of the scheme. Section 5 analyzes the security of this scheme. Section 6 evaluates the performance of this scheme. We give our conclusions in Section 7.

2 Preliminaries

2.1 Prime Order Bilinear Maps

Let $\mathbb{G}_1$ and $\mathbb{G}_T$ be two multiplicative cyclic groups of prime order p. Let h_1 and h_2 be the generator of $\mathbb{G}_1$ and e be a efficient computable bilinear map $e : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_T$. A bilinear map e has the following properties:

1) Bilinearity: $\forall \eta, \theta \in Z_N$, $e(h_1^{\eta}, h_2^{\theta}) = e(h_1, h_2)^{\eta\theta}$;

2) Non-degeneracy: $e(h_1, h_2) \neq 1$;

3) Computable: There exists an efficient algorithm could compute $e(h_1, h_2)$.

2.2 Access Structure

Definition 1. *(Access Structure [1]). Let $\mathcal{P}$ be a set of attributes. A collection $\mathbb{A}$ is monotone if $\forall \mathcal{P}_a, \mathcal{P}_b \subseteq 2^{\mathcal{P}}$ and $\mathcal{P}_b \subseteq \mathcal{P}_a, \exists C \subseteq 2^{\mathcal{P}}$ and $\mathcal{P}_b \subseteq \mathcal{P}_c$, then $\mathcal{P}_c \in \mathbb{A}$. A monotone access structure is a monotone collection $\mathbb{A}$ of non-empty subsets of $\mathcal{P}, i.e., \mathbb{A} \subseteq 2^{\mathcal{P}} \backslash \{\emptyset\}$. There is an attribute set $\mathcal{P}_S \subseteq \mathcal{P}$, $\mathcal{P}_S$ is called authorized set if $\mathcal{P}_S \in \mathbb{A}$, otherwise called unauthorized set. $\mathbb{A}$ is satisfied by authorized $\mathcal{P}_S$ and is not satisfied by unauthorized $\mathcal{P}_S$.*

Definition 2. *(Linear Secret Sharing Scheme). Let p be a prime, then the secret sharing scheme $\prod$ over a set of parties P is called linear (over Z_p) if P have two properties:*

1) There exists a vector(over Z_p), each party has the shares to constitute this vector.

2) M is a share-generating matrix which has l rows and n columns according to $\prod$, M_i is the ith row of M. The ith row of M is connected with a party $\rho(i)(i = 1, 2, \cdots, l)$ where $\rho : \{1, 2, \cdots, l\} \to Z_p$. Select a random vector $\overrightarrow{y} = (s, y_2, \cdots, y_n)^T \in Z_p^n$, where s represents the secret that the data owner wants to share, the l shares of secret s are expressed as $M\overrightarrow{y}$ which is the share for a $\rho(i)$.

Linear reconstruction: Define a set $I = \{i | \rho(i) \in S$, S is an authorized set $\}$. There exists a set of constants $\{\omega_i \in Z_p\}_{i \in I}$ satisfy $\sum_{i \in I} \omega_i \lambda_i = s$, where λ_i is any valid shares from $\prod$.

2.3 Complexity Assumption

Definition 3. *(DBDH). Let $\mathbb{G}_1$, $\mathbb{G}_T$ be multiplicative cyclic groups with prime order p according to a security parameter λ and g is the generator of $\mathbb{G}_1$. Let $e : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_T$ be a bilinear map [28], $\eta, \theta, \delta \in Z_p$ and $Q \in \mathbb{G}_T$ are selected randomly. For any PPT algorithm $\mathcal{T}$, there exists a negligible function negl(.), such that:*

$$|\Pr[\mathcal{T}(g, g^{\eta}, g^{\theta}, g^{\delta}, Z = e(g, g)^{\eta\theta\delta}) = 1] - \Pr[\mathcal{T}(g, g^{\eta}, g^{\theta}, g^{\delta}, Z = Q) = 1]| \leq negl(\lambda).$$

3 System and Security Model

3.1 System Model

The structure considers a smart grid access control system that involves five entities: Remote Terminal Unit (RTU), Cloud Data Center (CDC), Control Center (CC), User and Third-Party Auditor (TPA) as below. As it is shown in Figure 3.

- RTU: RTU comes from the substation, which responsible for collecting user data, encrypting the collected data and sending it to the CDC, while RTU is responsible for updating and deleting files.
- CDC: CDC is distributed in different geographical areas. It consists of a Data Service Manager (DSManager) and database. DSManager is semi-trusted and runs programs and related access protocols faithfully, but may be curious about the data content. The database only for storing the corresponding data file.
- CC: Based on the actual application of the smart grid, CC contains a key distribution authority (KDA). KDA is semi-trusted or malicious and capable of generating user secret keys.
- User: User can access the corresponding data when the access condition is met. And it may be malicious. The communication channel between all participants is considered secure.
- TPA: TPA is a third-party entity whose main task is to trace malicious entities and centers to guard against key abuse and accountability to the center.

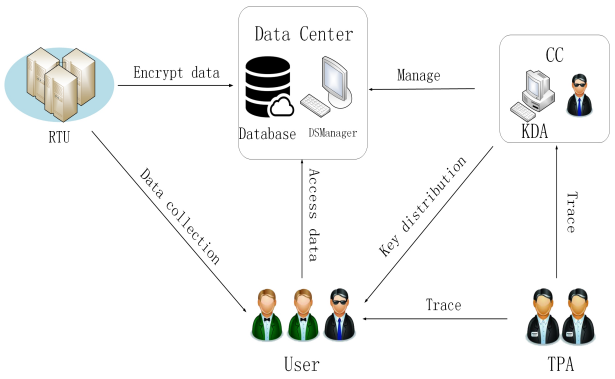


Figure 3: Interaction between entities in smart grid

3.2 BTA-CP-ABE Scheme

A black-box traceable and accountable CP-ABE scheme is similar to traditional ABE. For a given $\epsilon - useful$ decoder box $\mathbb{D}$, the creator of the box can be found through the trace algorithm. However, CP-ABE differs from traditional ABE in that the key generation is replaced by a protocol that can be used to implement the tracing feature. This scheme includes five algorithms and a protocol, as listed below.

Setup $(\lambda, \mathcal{U}) \rightarrow (PK, MSK)$**.** The input parameters are a security parameter λ and attribute universe $\mathcal{U}$. The output parameters are public key PK and master secret key MSK.

KeyGen $(PK, MSK, S, ID) \rightarrow SK$**.** This is an interactive protocol between the authority and user. The public input parameters are PK, attributes set S and user ID. The private input parameter is MSK. The end result of this protocol is secret key SK about ID and S.

Encrypt$(PK, m, (M, \rho)) \rightarrow CT$: The input parameters are PK, a message m, and an access structure $\mathbb{A}$ for attribute universe $\mathcal{U}$. The output parameter is a ciphertext CT.

Decrypt $(PK, SK, CT) \rightarrow m$ **or** $\perp$**.** The input parameters are key pair (PK, SK), and CT. If access policy $\mathbb{A}$ is satisfied by the attribute set S, it returns message m. If not, it returns $\perp$.

KCheck $(PK, SK) \rightarrow true$ **or** $false$**.** The input parameters are PK, and SK, and it returns $true$ if SK is well-formed, otherwise it returns $false$.

Trace $(PK, SK, \mathbb{D}) \rightarrow User$ **or** $Authority$**.** The input parameters are PK, SK which is pass **KCheck** algorithm, and an $\epsilon - useful$ black-box $\mathbb{D}$, it returns $Authority$ or $User$ means $\mathbb{D}$ is generated by authority or user.

Definition 4. *($\epsilon - Useful$ Decoder Box). Define a decoder black-box $\mathbb{D}$ that includes a value ϵ and a pair(t_k, t_c), where ϵ is non-negligible probability and (t_k, t_c) represents a secret token and ciphertext token respectively. This decoder black-box $\mathbb{D}$ can decrypt ciphertexts by t_k under t_c with probability at least ϵ(called $\epsilon - Useful$, and $\epsilon = 1/f(\lambda)$ for some polynomial f), that is [27]:*

$$\Pr[m \leftarrow \mathbb{D}(\boldsymbol{Encrypt}(PK, m, \mathbb{A}))] \geq \epsilon.$$

The abovementioned decryption black box reflects most of the real situation. Such a black box includes the decryption black-box similar to the key-like decryption black-box for sale mentioned in [22] and the decryption black-box "found in the wild". Once a black-box is found to decrypt ciphertexts which is associated with the token, we can regard it as a $\epsilon - Useful$ Decoder black-box.

3.3 Security Model

3.3.1 The IND-CPA Game

The $IND-CPA$ game for BTA-CP-ABE scheme is similar to that in the CP-ABE scheme but with the exception that key generation process is a interactive protocol. The game works as follows, which are played between a challenger $\mathcal{C}$ and an adversary $\mathcal{A}$.

Setup. The challenger runs **Setup** algorithm to generate a key pair (PK, MSK) and sends PK to $\mathcal{A}$.

Phase 1. In this phase, $\mathcal{A}$ adaptively queries $\mathcal{C}$ for secret keys which contains attribute sets and user ID $\{(ID_i, S_i)\}_{i \in q_1}$. Note that the secret keys obtained by any two identical (ID, S) are the same. For each (ID_i, S_i), $\mathcal{A}$ obtains secret key SK_i which is generated by key generation protocol with $\mathcal{C}$.

Challenge. When Phase 1 is over, $\mathcal{A}$ declares two equal length messages m_0, m_1, and an access structure $\mathbb{A}^*$. The challenge $\mathcal{C}$ flips a random coin $\sigma \in \{0,1\}$ and sends CT^* to $\mathcal{A}$, where CT^* is generated by **Encrypt** algorithm. Note that $\mathbb{A}^*$ is not satisfied by $\{S_i\}_{i\in q_1}$(associated with ID_i).

Phase 2. $\mathcal{A}$ queries secret keys with $\mathcal{C}$ adaptively in this phase, where the sets of attribute $\{(ID_i, S_i)\}_{i\in[q_1+1,q]}$ are not satisfy $\mathbb{A}^*$. $\mathcal{A}$ interacts with $\mathcal{C}$ for each (ID_i, S_i) are same as Phase 1.

Guess. $\mathcal{A}$ makes a guess σ', and $\mathcal{A}$ will succeed if $\sigma' = \sigma$.

The probability of $\sigma' = \sigma$ is $Pr[\sigma' = \sigma]$. And the advantage of $\mathcal{A}$ in this game is defined as: $Adv_{\mathcal{A}}^{IND-CPA}(\lambda) = |\Pr[\sigma' = \sigma] - 1/\ 2|$.

Definition 5. If the advantage of $\mathcal{A}$ winning the game is negligible in PPT, the BTA-CP-ABE scheme is IND-CPA secure.

3.3.2 The Dishonest Authority Game

The dishonest authority game works as follows. And $\mathcal{A}$(act as adversarial authority) try to create an $\epsilon-useful$ decoder box $\mathbb{D}^*$ that will frame the challenger $\mathcal{C}$(act as a honest user).

Setup. $\mathcal{A}$ generates public key PK, and sends PK, a user's pair (ID, S) to challenger $\mathcal{C}$. The challenger checks the PK with (ID, S) whether is well-formed or not, if not, it will abort.

KeyGen. $\mathcal{C}$ obtains the secret key SK which is generated by key generation protocol with $\mathcal{A}$. $\mathcal{C}$ runs **KCheck** algorithm, it aborts if algorithm returns $false$.

Query. $\mathcal{A}$ adaptively chooses ciphertexts CT and makes decryption queries. Then $\mathcal{C}$ sends m to $\mathcal{A}$, where m is decrypted by **Decrypt** algorithm.

Output. $\mathcal{A}$ uses the ID^* and attribute $S_{\mathbb{D}^*}$ to generate a decoder box $\mathbb{D}^*$, and will succeed if **Trace** algorithm returns $User$.

If trace algorithm returns user, then $\mathcal{A}$ has a advantage whose probability is: $\Pr[\textbf{Trace}(PK, SK, \mathbb{D}^*)=User]$

Definition 6. If the advantage of $\mathcal{A}$ winning the game is negligible in PPT, the BTA-CP-ABE scheme is Dishonest Authority secure.

3.3.3 The Dishonest User Game

In the dishonest user game, $\mathcal{A}$(act as adversarial user) may attempt to create an $\epsilon-useful$ decoder box $\mathbb{D}^*$ that will frame $\mathcal{C}$(act as authority). The process of game is as follows.

Setup. $\mathcal{C}$(act as authority)runs **Setup** algorithm, and sends the public key PK to $\mathcal{A}$.

KeyGen. $\mathcal{A}$ sends $\{(ID_i, S_i)\}_{i\in q}$ to $\mathcal{C}$ for the complete secret keys, then $\mathcal{A}$ obtains secret key SK_i which is generated by key generation protocol with $\mathcal{C}$.

Output. $\mathcal{A}$ uses the ID^* and attribute $S_{\mathbb{D}^*}$ to generate a decoder box $\mathbb{D}^*$, and will succeed if **Trace** algorithm returns $Authority$.

If the trace algorithm returns the authority, then $\mathcal{A}$ has a advantage whose probability is: $\Pr[\textbf{Trace}(PK, SK, \mathbb{D}^*)=Authority]$.

Definition 7. If the advantage of $\mathcal{A}$ winning the game is negligible in PPT, the BTA-CP-ABE scheme is Dishonest User secure.

4 Scheme Construction

The detailed construction of the proposed BTA-CP-ABE scheme will be explained in this section. The workflow of this scheme is illustrated in Figure 4.

4.1 System Initialization

At this stage, all of the entities will generate public parameters. They select two cyclic groups $\mathbb{G}$ and $\mathbb{G}_T$ with prime order p randomly. Let $PG = (\mathbb{G}, \mathbb{G}_T, g, p, e)$ be the public parameters, where g is the generator of $\mathbb{G}$ and $e : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_T$ is a bilinear map. Select a probabilistic symmetric encryption algorithm that could encrypt $\{0,1\}^*$ to Z_p^* by using key $\bar{k}$, and define the encryption and decryption algorithms as Enc and Dec respectively. Then the KDA generate PK and MSK by performing **Algorithm 1** and it retain MSK

Algorithm 1 Setup

Input: λ and $\mathcal{U}$
Output: PK and MSK

1: Randomly chooses $g_2 \in \mathbb{G}$,$\alpha, \beta \in Z_p$, computes $g_1 = g^{\alpha}$;
2: **for** i from 1 to $|\mathcal{U}|$ **do**
3: Randomly chooses $u_i \in Z_p$, computes $U_i = g^{u_i}$;
4: **end for**
5: **return**
6: $PK = \langle PG, g_1, g_2, e(g_1, g_2), \{U_i = g^{u_i}\}_{i\in\mathcal{U}}\rangle$;
7: $MSK = \langle \alpha, \beta, \bar{k}\rangle$;

4.2 Secret Key Generation

This phase is interactive. It requires users and the KDA to track the following protocols together.

- First, user selects $ID \in \{0,1\}^*$, and defines an attribute set S. Then user selects $t_2 \in Z_p$ and computes $R_1 = g^{t_2}, R_2 = g_2^{t_2}$. Then it runs a zero-knowledge protocol (ZKP) with KDA as Figure 5.

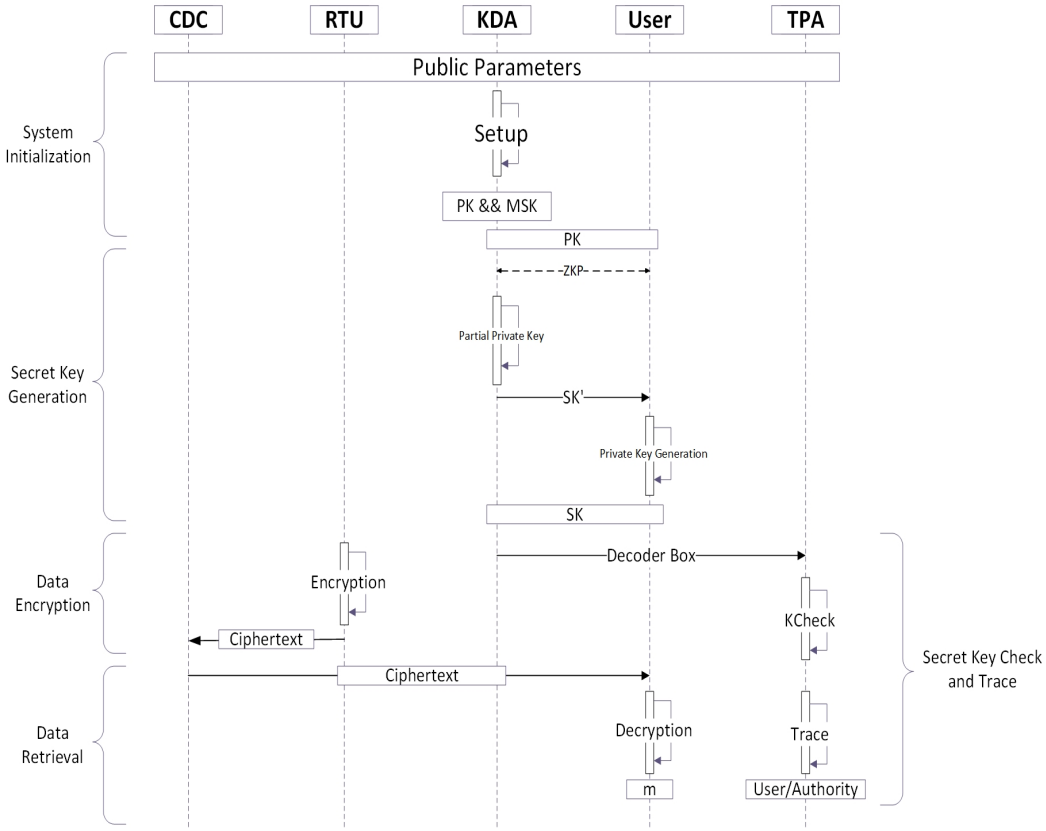


Figure 4: BTA-CP-ABE scheme

User		KDA
Randomly pick $l_1, l_2 \in Z_p$, compute $\bar{y}_1 = g^{l_1}$, $\bar{y}_2 = g_2^{l_2}$.		
	$\xrightarrow{\bar{y}_1, \bar{y}_2}$	
		Randomly chooses $c_1, c_2 \in Z_p$.
	$\xleftarrow{c_1, c_2}$	
Compute $s_1 = l_1 - t_2 c_1$, $s_2 = l_2 - t_2 c_2$.		
	$\xrightarrow{s_1, s_2}$	
		Verify if whether $\bar{y}_1 = g^{s_1} R_1^{c_1}$, $\bar{y}_2 = g_2^{s_2} R_2^{c_2}$ or not, if so accept R_1, R_2. Otherwise reject R_1, R_2.

Figure 5: A zero-knowledge protocol (ZKP) with KDA

- KDA outputs $\perp$ if the proof fails. Otherwise, it randomly compute $\bar{x} = Enc_{\bar{k}}(ID)$. Note that the value $\bar{x}$ has the same distribution as a random element in Z_p^*. KDA picks $r_1, t_1 \in Z_p$ randomly, and compute the partial private key $SK' = \langle S, \bar{D}_1, \bar{D}'_2, D'_1, D'_2, D'_3, D'_4, D'_5, \{\widetilde{D}'_i\}_{i \in S} \rangle$, where $\bar{D}_1 = \bar{x}$, $\bar{D}'_2 = t_1$, $D'_1 = g_2^{\alpha + r_1}$, $D'_2 = g^{r_1}$, $D'_3 = (R_1 g^{t_1})^{r_1}$, $D'_4 = (R_1 g^{t_1})^{r_1 \beta}$, $D'_5 = (R_2 g_2^{t_1})^{r_1}$, $\{\widetilde{D}'_i = U_i^{(\bar{x}+\beta) r_1}\}_{i \in S}$. Then KDA sends the SK' to user.

- User randomly chooses $r_2 \in Z_p$ and computes $\bar{D}_2 = \bar{D}'_2 + t_2$, $D_1 = D'_1 \cdot g_2^{r_2}$, $D_2 = D'_2 \cdot g^{r_2}$, $D_3 = D'_3 \cdot (g^{\bar{D}_2})^{r_2}$, $D_4 = D'_4 \cdot (g^{\bar{D}_2})^{\beta r_2}$, $D_5 = D'_5 \cdot (g_2^{\bar{D}_2})^{r_2}$, $\{\widetilde{D}_i = (\widetilde{D}'_i \cdot U_i^{(\bar{x}+\beta) r_2})^{\bar{D}_2}\}_{i \in S}$. Let $r = r_1 + r_2, t_k = t_1 + t_2$, then it gets the decryption key $SK = \langle S, \bar{D}_1, \bar{D}_2, D_1, D_2, D_3, D_4, D_5, \{\widetilde{D}_i\}_{i \in S} \rangle$, where $\bar{D}_1 = \bar{x}$, $\bar{D}_2 = t_k$, $D_1 = g_2^{\alpha + r}$, $D_2 = g^r$, $D_3 = g^{r t_k}$, $D_4 = g^{\beta r t_k}$, $D_5 = g_2^{r t_k}$, $\{\widetilde{D}_i = U_i^{(\bar{x}+\beta) r t_k}\}_{i \in S}$.

4.3 Data Encryption

At this stage, user data is encrypted by RTU through performing **Algorithm 2**. After RTU obtains the ciphertext, it uploads it to the CDC.

4.4 Data Retrieval

In this phase, users can employ the secret key to execute the **Algorithm 3** for data retrieval and access data.

Algorithm 2 Encryption

Input: PK, m, (M, ρ)
Output: CT

Randomly choose $t_c \in Z_p$, $s \in Z_p$, a vector $\overrightarrow{y} = (s, y_2, \cdots, y_n) \in Z_p^n$;
for j from 1 to l **do**
  Choose $\tau_j \in Z_N$ for each row M_j of M randomly;
  Compute $C_{j,1} = g^{\lambda_j} U_{\rho(j)}^{-t_c\tau_j}, C_{j,2} = g^{t_c\tau_j}$;
end for
return

$$CT = \langle C = m \cdot e(g_1, g_2)^s, \bar{C} = t_c, C_1 = g^s, C_2 = g^{st_c}, \{C_{j,1} = g^{\lambda_j} U_{\rho(j)}^{-t_c\tau_j}, C_{j,2} = g^{t_c\tau_j}\}_{j=1}^{l} \rangle;$$

Algorithm 3 Decryption

Input: PK, SK, CT
Output: m, if $\bar{C} \neq \bar{D}_2$ and S satisfy the (M, ρ); $\perp$, otherwise

Compute constants $\omega_j \in Z_N$ such that $\sum_{\rho(j)\in S} \omega_j M_j = (1, 0, \cdots, 0)$;
Compute

$$E = \left(\frac{\prod_{\rho(j)\in S} \left(e((D_3)^{\bar{D}_1} D_4, C_{j,1}) e(\widetilde{D}_{\rho(j)}, C_{j,2})\right)^{\omega_j}}{e((D_3)^{\bar{D}_1} D_4 D_5, C_1) e(D_2, C_2)^{-1}}\right)^{\frac{1}{\bar{C}-\bar{D}_2}};$$

if $\bar{C} \neq \bar{D}_2$ **then**
  Compute $F = e(D_1, C_1), m = \frac{C \cdot E}{F}$;
  return m;
else
  return $\perp$;
end if

Compute the correctness of decryption:

$$\begin{aligned} E &= \left(\frac{\prod_{\rho(j)\in S} \left(e((D_3)^{\bar{D}_1} D_4, C_{j,1}) e(\widetilde{D}_{\rho(j)}, C_{j,2})\right)^{\omega_j}}{e((D_3)^{\bar{D}_1} D_4 D_5, C_1) e(D_2, C_2)^{-1}}\right)^{\frac{1}{\bar{C}-\bar{D}_2}} \\ &= \left(\frac{\prod_{\rho(j)\in S} \left(e(g^{(\beta+\bar{x})rt_k}, g^{\lambda_j})\right)^{\omega_j}}{e(g,g)^{(\beta+\bar{k})rst_k} e(g_2, g)^{rst_k} e(g, g_2)^{-rst_c}}\right)^{\frac{1}{t_c-t_k}} \\ &= e(g, g_2)^{rs}; \\ F &= e(D_1, C_1) = e\left(g_2^{\alpha+r}, g^s\right), \\ m &= \frac{C \cdot E}{F} = \frac{m \cdot e(g_1, g_2)^s \cdot e(g, g_2)^{rs}}{e\left(g_2^{\alpha+r}, g^s\right)}; \end{aligned}$$

4.5 Secret Key Check and Trace

This stage includes two algorithms: A key check algorithm and a trace algorithm. A malicious user or authority may generate a decoder, which may illegally decrypt the ciphertext to leak private information. In order to trace the owner of the decoder, the TPA needs to execute the **Algorithm 4** to check whether the secret key is valid, and then preform the **Algorithm 5** to trace the malicious entity.

Algorithm 4 KCheck

Input: PK, SK
Output: $true$ or $false$

if SK is well-fromed **then**
  Check whether SK has the form of $\left\langle S, \bar{D}_1, \bar{D}_2, D_1, D_2, D_3, D_4, D_5, \left\{\widetilde{D}_i\right\}_{i\in S}\right\rangle$ and $\bar{D}_2, D_1, D_2, D_3, D_4, \{\widetilde{D}_i\}_{i\in S} \in G, \bar{D}_1 \in Z_p^*$;
  Compute $e(D_1, D_3) = e(g_1 D_2, D_5)$;
  for i from 1 to $|S|$ **do**
    Compute $e(\widetilde{D}_i, g) = e(D_3{}^{\bar{D}_1} D_4, U_i)$;
    return true;
  end for
else
  return false;
end if

Algorithm 5 Trace

Input: PK, SK,$\mathbb{D}$
Output: U or Authority

if KCheck(PK, SK)=$false$ **then**
  return $\perp$;
else
  Define a counter $\xi = 0$;
  for i from 1 to $L(L = \lambda/\epsilon)$ **do**
    Set $t_c = t_k$;
    Construct an access structure (M,ρ) that can be satisfied by the box attribute set $S_{\mathbb{D}}$;
    Run **Algorithm 2** to generate a ciphertext CT;
    Use the black-box $\mathbb{D}$ to decrypt CT, and gets m';
    if $m' = m$ **then**
      Set ξ=ξ+1;
    end if
  end for
  if $\xi = 0$ **then**
    Extract ID from $Dec_{\bar{k}}(\bar{D}_1) = Dec_{\bar{k}}(Enc_{\bar{k}}(ID))$;
    return $User$;
  else
    return $Authority$;
  end if
end if

At this scheme, token t_k marks the secret key, which is used to trace the malicious entity, and t_k is composed of two parts: One generated by the user and one by the KDA. This effectively weakens the KDA's complete control of the key. Once a malicious entity conducts an illegal

transaction of the private key on both entities. In the **Algorithm 5**, if $t_k = t_c$, the secret key (associated with t_k) cannot correctly decrypt the ciphertext (associated with t_c). Therefore, CT cannot be decrypted correctly if the input SK is used to generate $\mathbb{D}$. If it decrypts the ciphertext CT correctly, then the creator of $\mathbb{D}$ is KDA. The message space size is the exponential level of λ, so the probability that $\mathbb{D}$ can correctly guess m is negligible.

5 Security Analysis

Lemma 1. *(IND-CPA Security): If the advantage of an adversary in wining the IND-CPA game for the scheme is negligible, then the scheme is IND-CPA security.*

Proof: If there exists a adversary $\mathcal{A}$ who can break the $DBDH$ assumption with a non-negligible advantage ε, then a polynomial-time simulator $\mathcal{C}$ can be built to solve the $DBDH$ problem.

Setup. $\mathcal{C}$ randomly chooses $\beta' \in Z_p$, and implicitly sets α, β as a, β',respectively. For each attribute $x \in \mathcal{U}$, select a element $u'_x \in Z_p$ randomly. Then public key is set as $PK = \langle PG, g_1 = g^a, g_2 = g^b, \{U_i = g^{u'_x}\}_{i\in\mathcal{U}}\rangle$, and is sent to $\mathcal{A}$.

Phase 1. In this phase, $\mathcal{A}$ adaptively queries $\mathcal{C}$ for secret keys which contains attribute sets and user ID $\{(ID_i, S_i)\}_{i\in q_1}$. For a query on (ID_i, S_i), $\mathcal{A}$ obtains secret key SK_i which is generated by key generation protocol with $\mathcal{C}$:

1) $\mathcal{C}$ and $\mathcal{A}$ interact through ZKP. Abort the proof if fails. Otherwise $\mathcal{C}$ receives $R_1 = g^{t_2}, R_2 = g_2^{t_2}$.
2) $\mathcal{C}$ chooses $t, r' \in Z_p$ randomly, and sets $r_1 = -a + r', t_k = t$ implicitly. Then computes simulatively a partial secret key as the for well-formed secret key $\widehat{SK} = \langle S, \widehat{\bar{D}}_1, \widehat{\bar{D}}_2, \widehat{D}_1, \widehat{D}_2, \widehat{D}_3, \widehat{D}_4, \widehat{D}_5, \{\widehat{\widetilde{D}}_i\}\rangle$, where $\widehat{\bar{D}}_1 = \bar{x}, \widehat{\bar{D}}_2 = t_k, \widehat{D}_1 = g_2^{\alpha+r_1}, \widehat{D}_2 = g^{r_1}, \widehat{D}_3 = g^{r_1 t_k}, \widehat{D}_4 = g^{\beta r_1 t_k}, \widehat{D}_5 = g_2^{r_1 t_k}, \{\widehat{\widetilde{D}}_i = U_i^{(\bar{x}+\beta)r_1 t_k}.\}$ $\mathcal{C}$ rewinds $\mathcal{A}$ to obtain t_2. Then computes partial secret key:

$$\begin{aligned}
&\bar{D}'_2 = \widehat{\bar{D}}_2 - t_2 = t_k - t_2,\\
&D'_1 = \widehat{D}_1 = g_2^{\alpha+r_1},\\
&D'_2 = \widehat{D}_2 = g^{r_1},\\
&D'_3 = \widehat{D}_3 = g^{r_1 t_k} = (g^{t_2} g^{t_k - t_2})^{r_1},\\
&D'_4 = \widehat{D}_4 = g^{\beta r_1 t_k} = (g^{t_2} g^{t_k - t_2})^{r_1 \beta'},\\
&D'_5 = \widehat{D}_5 = g_2^{r_1 t_k} = (g_2^{t_2} g_2^{t_k - t_2})^{r_1},
\end{aligned}$$

For $i \in \mathcal{U}$, it sets:

$$\widetilde{D}'_i = \widehat{\widetilde{D}}_i^{\frac{1}{t_k}} = U_i^{(\bar{x}+\beta')r_1}.$$

Let $t_1 = t_k - t_2$. The partial secret key is set as $SK' = \langle S, \bar{D}_1, \bar{D}'_2, D'_1, D'_2, D'_3, D'_4, D'_5, \{\widetilde{D}'_i\}\rangle$, where $\bar{D}_1 = \bar{x}, \bar{D}'_2 = t_1, D'_1 = g_2^{\alpha+r_1}, D'_2 = g^{r_1}, D'_3 = (R_1 g^{t_1})^{r_1}, D'_4 = (R_1 g^{t_1})^{r_1\beta'}, D'_5 = (R_2 g_2^{t_1})^{r_1}, \{\widetilde{D}'_i = U_i^{(\bar{x}+\beta')r_1}\}$.
Therefore, the simulator $\mathcal{C}$ can successfully simulate the partial private key.

3) $\mathcal{A}$ random choose $r_2 \in Z_p$, and computes:

$$\begin{aligned}
&\bar{D}_2 = t_1 + t_2 = t_k\\
&D_1 = D'_1 \cdot g_2^{\alpha+r_2} = g_2^{\alpha+r_1} g_2^{r_2} = (g^b)^{a-a+\widetilde{r}+r_2}\\
&D_2 = D'_2 \cdot g^{r_2} = g^{-a+\widetilde{r}+r_2}\\
&D_3 = D'_3 \cdot (g^{\bar{D}_2})^{r_2} = g^{(-a+\widetilde{r}+r_2)t_k}\\
&D_4 = D'_4 \cdot (g^{\bar{D}_2})^{r_2\beta'} = g^{(-a+\widetilde{r}+r_2)t_k\beta'}\\
&D_5 = D'_5 \cdot (g_2^{\bar{D}_2})^{r_2} = g_2^{(-a+\widetilde{r}+r_2)t_k}
\end{aligned}$$

$$\left\{\widetilde{D}_j = (\widetilde{D}'_j \cdot U_j^{(\bar{x}+\beta')r_2})^{\bar{D}_2} = U_j^{(\bar{x}+\beta')(-a+\widetilde{r}+r_2)t_k}\right\}_{j\in S_i}$$

The complete secret key is set as $SK = (S_i, \bar{D}_1 = \bar{x}, \bar{D}_2, D_1, D_2, D_3, D_4, D_5, \{\widetilde{D}_j\}_{i\in S_i})$, then $\mathcal{C}$ sends SK to $\mathcal{A}$.

Challenge: When Phase 1 is over, $\mathcal{A}$ declares two equal length messages m_0, m_1, and an access structure $\mathbb{A}^* = (M^*, \rho^*)$. Note that $\{S_i\}_{i\in q_1}$(associated with ID_i) is not satisfy the $\mathbb{A}^*$. $\mathcal{C}$ flips a random coin $\sigma \in \{0, 1\}$, and implicitly sets $t_c = t$, $s = c$, it chooses a vector $\overrightarrow{y'} = (1, y'_2, \cdots, y'_{n^*}) \in Z_p^{n^*}$ and computes:

$$\begin{aligned}
&C = m \cdot e(g_1, g_2)^s = m \cdot e(g, g)^{abc},\\
&\bar{C} = t_c = t,\\
&C_1 = g^s = g^c,\\
&C_2 = g^{st_c} = (g^c)^t.
\end{aligned}$$

For each row M_j^* of M^*, it randomly chooses $\delta_i \in Z_p$, then computes:

$$\{C_{j,1} = g^{\lambda_j} U_{\rho(j)}^{-t_c\tau_j} = g^{c\lambda_j} U_{\rho^*(j)}^{-t\delta_j}, C_{j,2} = g^{t_c\tau_j} = g^{t\delta_j}\}_{j=1}^{l^*}.$$

The ciphertext sets as $CT^* = (C, \bar{C}, C_1, C_2, \{C_{j,1}, C_{j,2}\})$, then $\mathcal{C}$ sends CT^* to $\mathcal{A}$.

Phase 2. In this phase, $\mathcal{A}$ queries secret keys with $\mathcal{C}$ adaptively, where the sets of attribute $\{(ID_i, S_i)\}_{i\in[q_1+1,q]}$ are not satisfy $\mathbb{A}^*$. Secret key generation is same as Phase 1.

Guess. $\mathcal{A}$ outputs its guess σ' of σ, $\mathcal{A}$ will succeed if $\sigma' = \sigma$.

The advantage of $\mathcal{C}$ as follows. According to the assumption that $\mathcal{A}$ can break the scheme with a non-negligible advantage ε, then the probability of $\mathcal{A}$ in guessing the encrypted message correctly to be $|Pr[\sigma' = \sigma]|Z = e(g, g)^{abc}]| = 1/2 + \varepsilon$. $\mathcal{A}$ does not know anything about CT^* because Z is random. And Z is used to encrypt m. Then the probability of $\mathcal{A}$ in guessing the m correctly is $|Pr[\sigma' = \sigma]|Z \neq e(g, g)^{abc}]| = 1/2$. $\mathcal{C}$ could solve $DBDH$ problem, and the probability is $|Pr[\sigma' = \sigma]|Z = e(g, g)^{abc}]| - Pr[\sigma' = \sigma]|Z \neq e(g, g)^{abc}]| = \varepsilon$.

Lemma 2. *(Dishonest Authority Security): If the advantage of an adversary in wining the dishonest authority game for the scheme is negligible, then the scheme is dishonest authority security.*

Proof. If $\mathcal{A}$ let the trace algorithm return $User$ with challenger $\mathcal{C}$ as follows, then $\mathcal{A}$ wins the dishonest authority game.

Setup. $\mathcal{A}$ generates public key PK, and sends PK, a user's pair (ID, S) to $\mathcal{C}$. $\mathcal{C}$ aborts if PK and (ID, S) are not well-formed.

KeyGen: $\mathcal{C}$ interacts with $\mathcal{A}$ to obtain secret key SK corresponding to a token t_k for the pair (ID, S) through key generation protocol. $\mathcal{C}$ obtains secret key SK and runs **KCheck** algorithm ensure that the key is well-formed, it aborts if check fails.

Query. $\mathcal{A}$ adaptively chooses ciphertexts CT(associated with the token t_c) and makes decryption queries. $\mathcal{C}$ checks whether CT is well-formed, if not, it aborts. $\mathcal{C}$ runs **Decrypt** algorithm and sends the result to $\mathcal{A}$.

Output: $\mathcal{A}$ outputs an $\epsilon - useful$ decoder box $\mathbb{D}^*$ for user ID^* and attribute $S_{\mathbb{D}^*}$, and will succeed if **Trace** algorithm returns $User$.

If **Trace**$(PK, SK, \mathbb{D}^*)$=$User$, it means that the counter $\xi = 0$ after trace algorithm has finished running. Assuming authority generates the decoder box $\mathbb{D}$, the probability that the counter $\xi = 0$ after the end of the trace algorithm is negligible for the following reasons.

The probability that malicious authority can obtain the token t_k is negligible. Due to KG-Anonymity security (proven in Lemma 5), the possibility that $\mathcal{A}$ can obtain t_k from the private key SK during the private key generation phase is negligible. $\mathcal{A}$ adaptively selects the ciphertext for decryption in the decryption query phase. Only a well-formed ciphertext can be correctly decrypted, and a well-formed ciphertext only can be decrypted by a well-formed secret key for obtaining same result, so the probability that $\mathcal{A}$ obtains the t_k is negligible.

The authority may extract the token t_k with negligible probability. If decoder box $\mathbb{D}$ is considered stateless, the probability of keeping ξ constant is at most $1 - \epsilon$ in the trace algorithm. The probability is negligible, and it can be computed as follows:

$$\Pr[\xi = 0] \leq (1-\epsilon)^L \approx \exp(-\epsilon\lambda) exp(-\epsilon \cdot \lambda/\epsilon) = exp(-\lambda).$$

Hence, the advantage of $\mathcal{A}$ in wining the game is negligible. □

Lemma 3. *(Dishonest User Security): If the advantage of an adversary in wining the dishonest user game for the scheme is negligible, then the scheme is dishonest user security.*

Proof. Assume an adversary $\mathcal{A}$ can break the dishonest user security of this scheme. Then another adversary $\mathcal{B}$ can be constructed according to $\mathcal{A}$, then the adversary $\mathcal{B}$ can break the $IND - CPA$ security.

Setup. The challenger $\mathcal{C}$ generates PK through **Setup** algorithm. Then $\mathcal{B}$ obtains PK and sends it to $\mathcal{A}$.

Phase 1. $\mathcal{A}$ adaptively queries the secret key on (ID, S). $\mathcal{B}$ interacts with $\mathcal{A}$ and $\mathcal{C}$ respectively to generate a well-formed SK by (ID, S, t_k) through key generation protocol, where $\mathcal{B}$ acts as a middleman. $\mathcal{B}$ sends anything received from $\mathcal{A}$ to $\mathcal{C}$, and sends anything received from $\mathcal{C}$ to $\mathcal{A}$, and finally $\mathcal{A}$ gets the private key SK.

Challenge. $\mathcal{B}$ obtains a black-box $\mathbb{D}^*$ and SK by (ID^*, S^*, t_k^*). Then it sends two equal length messages m_0, m_1, and an access structure $\mathbb{A}^*$ to $\mathcal{C}$. $\mathcal{C}$ makes $t_c^* = t_k^*$ and flips a random coin $\sigma \in \{0, 1\}$. Note that $\mathbb{A}^*$ cannot be satisfies by any queried attribute set (ID^*, S^*). Then $\mathcal{C}$ runs **Encrypt** algorithm to generate a challenge ciphertext CT^*, and sends CT^* to $\mathcal{B}$.

Phase 2. $\mathcal{B}$ gets decoder box $\mathbb{D}^*$ about (ID^*, S^*), so this phase is omitted.

Guess. Trace algorithm will return $Authority$. This means at least in one iteration the decoder box $\mathbb{D}^*$ outputs m'=m and CT^* is encrypted by t_k^*. So the $\mathcal{A}$ will succeed in dishonest user game. Note that t_k^* is the token of the input SK. $\mathcal{B}$ uses the $\mathbb{D}$ to decrypt CT^*, the probability of that $\mathbb{D}$ output m_σ^* is $1/L(L = \lambda/\varepsilon)$ which is non-negligible. $\mathcal{B}$ receives m_σ^*, if $m_\sigma^* = m_0$, $\mathcal{B}$ returns its guess as $\sigma = 0$. Otherwise if $m_\sigma^* = m_1$, it returns $\sigma = 1$. If adversary guesses σ correctly, the probability is non-negligible.

Hence, The probability of $\mathcal{B}$ break $IND - CPA$ security is non-negligible if $\mathcal{A}$ could break the dishonest user security. The BTA-CP-ABE scheme is dishonest user secure if and only if scheme is $IND - CPA$ secure. □

Lemma 4. *Lemma 4(KG-Anonymity Security): If the key generation protocol satisfies KG-Anonymity Security, the interactive protocol is secure.*

Proof. User obtain a corresponding secret key SK with (ID, S, t_k) through key generation protocol, where the token t_k is computed by t_1, t_2, and R_1, R_2 is computed with t_2 as R_1, R_2 without leaking the secret of t_2. ZKP protects t_2 from being leaked, that is, authority cannot obtain any information about t_k. The ZKP is mentioned in [33]. The probability in guessing t_k is negligible because of $t_k \in Z_p$. Therefore, the key generation protocol is satisfying the KG-Anonymity security. □

Table 1: Comparisons with other schemes

Schemes	Category	Access Structures	Group Order	Traceability	Authority Accountability
[17]	KP-ABE	LSSS	Composite	√	×
[30]	KP-ABE	Tree	Prime	√	√
[8]	CP-ABE	AND	Prime	White-box	×
[15]	CP-ABE	LSSS	Composite	Black-box	×
[21]	CP-ABE	LSSS	Composite	White-box	√
[23]	CP-ABE	LSSS	Prime	Black-box	×
[24]	CP-ABE	Tree	Prime	Black-box	×
[33]	CP-ABE	LSSS	Prime	White-box	√
[35]	CP-ABE	LSSS	Prime	White-box	√
Ours	CP-ABE	LSSS	Prime	Black-box	√

6 Performance Evaluation

In this section, we compare the function and performance of this scheme with other schemes. We mainly rely on the type of the category, access structure, group order, traceability, and authority accountability. In Table 1, the two schemes ([17] and [30]) are KP-ABE, which support user traceability, but the category is not suitable for smart grid. The schemes in [8, 15, 21, 23, 24, 33, 35] are CP-ABE, and in [21,33] and [35] tracing and authority accountability are combined. [15] and [21] scheme are based on the composite order, so the performance of these two schemes is worse than the others under the same conditions. Although the scheme in [8] is based on prime order groups, its access structure is AND gate which may not be as flexible compared to other schemes and it is white-box traceable. Both [23] and [24] schemes are constructed from prime order groups and support black-box traceability, but they cannot support the authority accountability function.

Table 2: Parameters and descriptions

Parameters	Descriptions
$\|\mathcal{U}\|$	Number of attributes in system
$\|\mathbb{G}\|$	Number of elements in group $\mathbb{G}$
$\|\mathbb{G}_T\|$	Number of elements in group $\mathbb{G}_T$
$\|S\|$	Number of attributes of user
l	Access policy length(number of attributes in access policy)
$\|I\|$	Minimum authorized set in decryption process(number of attributes that satisfy access policy)
E	Exponential operation in groups $\mathbb{G}$ and $\mathbb{G}_T$
P	Bilinear pairing operation

In the performance comparisons phase, we compare our schemes with [8, 23, 24, 33] and [35] are given as follows. The parameters are defined in Table 2. The comparisons mainly include public key length, secret key length, ciphertext length, encryption overhead and decryption overhead. According to the theoretical comparison re-

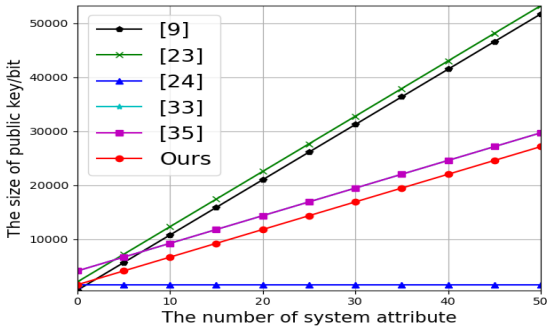


Figure 6: Size of public parameters

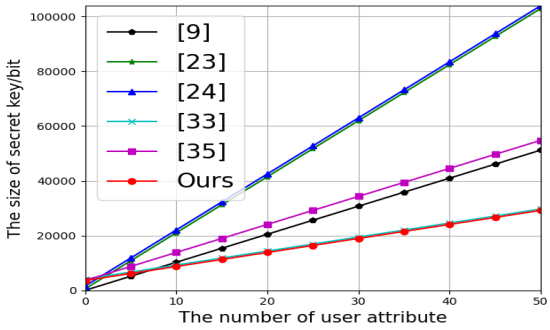


Figure 7: Size of private parameters

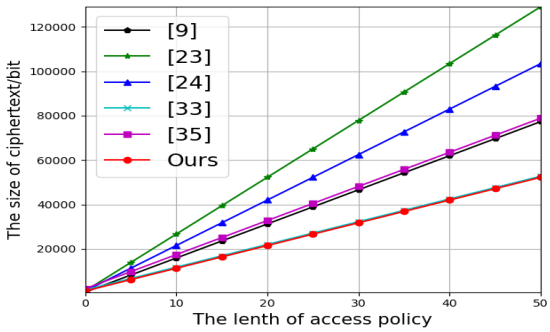


Figure 8: Size of ciphertexts

Table 3: Performance comparisons with other schemes

Schemes	Public Key Size	Secret Key Size	Ciphertext Size	Setup Cost	Encryption Cost	Decryption Cost
[8]	$2\|\mathcal{U}\|\|\mathbb{G}\| + \|\mathbb{G}_T\|$	$2\|S\|\|\mathbb{G}\|$	$3l\|\mathbb{G}\| + \|\mathbb{G}_T\|$	$(2\|\mathcal{U}\| + 1)E + P$	$(3l+1)E$	$2\|I\|P$
[23]	$(2\|\mathcal{U}\| + 3)\|\mathbb{G}\| + \|\mathbb{G}_T\|$	$(4\|S\|+1)\|\mathbb{G}\|$	$(5l+1)\|\mathbb{G}\| + \|\mathbb{G}_T\|$	$(2\|\mathcal{U}\| + 2)E + P$	$(7l+2)E$	$(5\|I\|+1)P$
[24]	$2\|\mathbb{G}\| + \|\mathbb{G}_T\|$	$(4\|S\|+1)\|\mathbb{G}\|$	$(4l+1)\|\mathbb{G}\| + \|\mathbb{G}_T\|$	$\|\mathcal{U}\|E + P$	$(5l+2)E$	$(4\|I\|+1)P$
[33]	$(\|\mathcal{U}\| + 2)\|\mathbb{G}\| + 6\|\mathbb{G}_T\|$	$(\|S\| + 5)\|\mathbb{G}\| + 3\|Z_p^*\|$	$(2l+2)\|\mathbb{G}\| + \|\mathbb{G}_T\|$	$(\|\mathcal{U}\|+8)E + P$	$(3l+3)E$	$5E + (2\|I\| + 3)P$
[35]	$\|\mathcal{U}\|\|Z_p^*\| + 7\|\mathbb{G}\| + \|\mathbb{G}_T\|$	$(2\|S\| + 4)\|\mathbb{G}\| + 3\|Z_p^*\|$	$(3l+3)\|\mathbb{G}\| + \|\mathbb{G}_T\|$	$(\|\mathcal{U}\|+3)E + P$	$(5l+3)E$	$4E + (3\|I\| + 2)P$
ours	$(\|\mathcal{U}\| + 2)\|\mathbb{G}\| + \|\mathbb{G}_T\|$	$(\|S\| + 5)\|\mathbb{G}\| + 2\|Z_p^*\|$	$(2l+2)\|\mathbb{G}\| + \|\mathbb{G}_T\|$	$(\|\mathcal{U}\|+1)E + P$	$(3l+3)E$	$E + (2\|I\| + 3)P$

sults of Table 3, the lengths of the secret key and ciphertext in our scheme are shorter than in other schemes, and fewer indexing operations and the pairing operations are required for the encryption and decryption overhead. Figures 6, 7, and 8 are the length comparison of each scheme more intuitively, and the proposed scheme has an advantage when it comes to the length of the secret key and ciphertext.

The experimental results are shown in Figures 9, 10, 11, and 12. The scheme is based on the JPBC library, and the experimental environment is a Windows 7 laptop with a 3.30 GHz Intel Core CPU and 4 GB RAM. We compared the [8, 23, 24, 33, 35] and our schemes in terms of the time consumption for setup, key generation, encryption and decryption. The result is the average value obtained after performing the experiment multiple times.

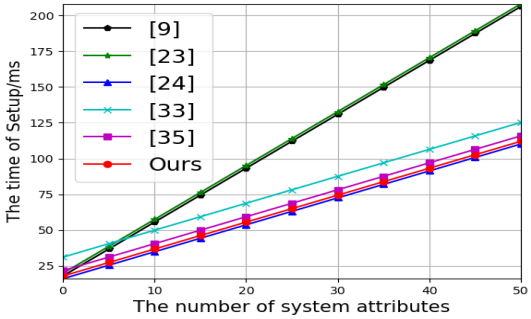


Figure 9: Time of setup

Figure 9 shows the linear relationship between $|\mathcal{U}|$ and the time of setup. Figure 10 shows the linear relationship between $|S|$ and the time of generating the secret key. Figure 11 shows the linear relationship between S in access policy length l and the time of generating ciphertext, and Figure 12 shows the linear relationship between $|I|$ and decryption time when correctly decrypted. Our scheme performs better than the other schemes. Our access policy is more flexible than other schemes, and this scheme is proven to be secure under the standard model.

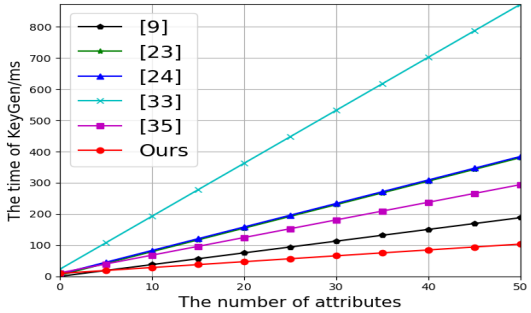


Figure 10: Time of key generation

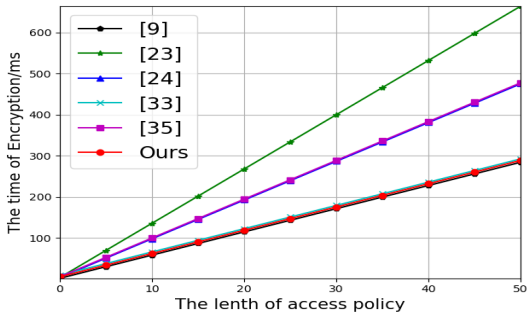


Figure 11: Time of encryption

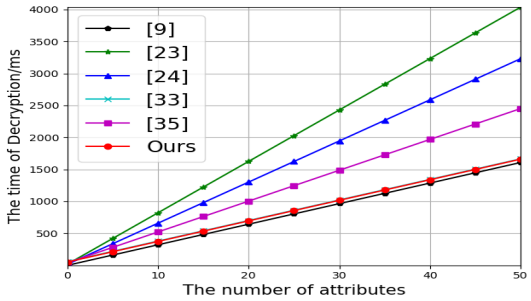


Figure 12: Time of decryption

Results show that our proposed scheme is more practical and efficient than others for smart grid applications.

7 Conclusion

In this study, a novel secure and accountable sharing framework for smart grids is proposed. This scheme can protect data security, as well as data privacy in smart grids. Our scheme achieves the traceability of malicious users and the accountability of malicious authorities using a token-based technology. In order to weaken the control of authority, we introduced a user-authority interaction protocol, which uses zero-knowledge proof to ensure that secret keys are not tampered by malicious authorities. The comparisons show that our solution is more secure and more efficient than other schemes. We hope to extend the system by adding features, such as large universe or user revocation, in the future.

Acknowledgments

This work was part supported by the National Cryptography Development Fund under grant (MMJJ20180209), Xi'an Science and Technology Plan Project No. 2020KJRC0109. Key Foundation of National Natural Science Foundation of China under grant (NO. U19B2021).

References

[1] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attribute-based encryption," in *IEEE Symposium on Security and Privacy*, pp. 321–334, 2007.

[2] R. Bobba, H. Khurana, M. Alturki, and F. Ashraf, "PBES: A policy based encryption system with application to data sharing in the power grid," in *The 4th International Symposium on Information, Computer, and Communications Security*, pp. 262–275, 2009.

[3] A. Bose, "Smart transmission grid applications and their supporting infrastructure," *IEEE Transactions on Smart Grid*, vol. 1, no. 1, pp. 11–19, 2010.

[4] Z. Fadlullah, N. Kato, R. Lu, X. Shen, and Y. Nozaki, "Towards secure targeted broadcast in smart grid," *IEEE Communications Magazine*, vol. 50, no. 5, pp. 150–156, 2012.

[5] V. Goyal, "Reducing trust in the pkg in identity based cryptosystems," in *The 27th Annual International Cryptology Conference*, pp. 430–447, 2007.

[6] V. Goyal, S. Lu, A. Sahai, and B. Waters, "Black-box accountable authority identity-based encryption," in *The 15th ACM Conference on Computer and Communications Security*, pp. 427–436, 2008.

[7] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data," in *The 13th ACM Conference on Computer and Communications Security (CCS'06)*, pp. 89–98, 2006.

[8] Y. Jiang, W. Susilo, Y. Mu, and F. Guo, "Ciphertext-policy attribute-based encryption against key-delegation abuse in fog computing," *Future Generation Computer Systems-The International Journal of Escience*, vol. 78, pp. 720–729, 2018.

[9] Y. J. Kim, M. Thottan, V. Kolesnikov, and W. Lee, "A secure decentralized data-centric information infrastructure for smart grid," *IEEE Communications Magazine*, vol. 48, no. 11, pp. 58–65, 2010.

[10] J. Lai and Q. Tang, "Making any attribute-based encryption accountable, efficiently," in *The 23rd European Symposium on Research in Computer Security*, pp. 527–547, 2018.

[11] A. Lee, T. L. Brewer, *Smart Grid Cyber Security Strategy and Requirements*, 2009. Corpus ID: 168831134.

[12] J. Li, K. Ren, and B.Zhu, "Privacy-aware attribute-based encryption with user accountability," in *The 12th Information Security Conference*, pp. 347–367, 2009.

[13] J. Li, K. Ren, and K. Kim, "A2BE: Accountable attribute-based encryption for abuse free access control," *Computer Science*, 2009. Corpus ID: 306392.

[14] X. Liang, R. Lu, X. Shen, X. Lin, and H. Zhu, "Securing smart grid: Cyber attacks, countermeasures, and challenges," *IEEE Communications Magazine*, vol. 50, no. 8, pp. 38–45, 2012.

[15] Z. Liu, Z. Cao, and D. S Wong, "Blackbox traceable CP-ABE: How to catch people leaking their keys by selling decryption devices on ebay," in *ACM Conference on Computer and Communications Security*, pp. 475–486, 2013.

[16] Z. Liu, Z. Cao, and D. S Wong, "White-box traceable ciphertext-policy attribute-based encryption supporting any monotone access structures," *IEEE Transactions on Information Forensics and Security*, vol. 8, no. 1, pp. 76–88, 2013.

[17] Z. Liu, Z. Cao, and D. S. Wong, "Fully collusion-resistant traceable key-policy attribute-based encryption with sub-linear size ciphertexts," in *The 10th International Conference on Information Security and Cryptology*, pp. 403–423, 2015.

[18] Z. Liu, Z. Cao, and D. S. Wong, "Traceable CP-ABE: How to trace decryption devices found in the wild," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 1, pp. 55–68, 2015.

[19] Z. Liu, S. Duan, P. Zhou, and B. Wang, "Traceable-then-revocable ciphertext-policy attribute-based encryption scheme," *Future Generation Computer Systems*, vol. 93, pp. 903–913, apr. 2019.

[20] J. Ning, Z. Cao, X. Dong, K. Liang, L. Wei, and K. R. Choo, "Cryptcloud+: Secure and expressive data access control for cloud storage," *IEEE Transactions on Services Computing*, vol. 14, no. 1, pp. 111–124, 2018.

[21] J. Ning, X. Dong, Z. Cao, and L. Wei, "Accountable authority ciphertext-policy attribute-based encryption with white-box traceability and public auditing in the cloud," in *The 20th European Symposium on Research in Computer Security*, pp. 270–289, 2015.

[22] J. Ning, X. Dong, Z. Cao, L. Wei, and X. Lin, "White-box traceable ciphertext-policy attribute-based encryption supporting flexible attributes," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 6, pp. 1274–1288, 2015.

[23] H. Qiao, H. Ba, H. Zhou, Z. Wang, J. Ren, and Y. Hu, "Practical, provably secure, and black-box traceable cp-abe for cryptographic cloud storage," *Symmetry-Basel*, vol. 10, no. 10, pp. 1–17, 2018.

[24] H. Qiao, J. Ren, Z. Wang, H. Ba, and H. Zhou, "Compulsory traceable ciphertext-policy attribute-based encryption against privilege abuse in fog computing," *Future Generation Computer Systems-The International Journal of Escience*, vol. 88, pp. 107–116, 2018.

[25] S. R. Rajagopalan, L. Sankar, S. Mohajer, and H. V. Poor, "Smart meter privacy: A utility-privacy framework," in *Smart Grid Communications*, pp. 190–195, 2011.

[26] S. Ruj and A. Nayak, "A decentralized security framework for data aggregation and access control in smart grids," *IEEE Transactions on Smart Grid*, vol. 4, no. 1, pp. 196–205, 2013.

[27] A. Sahai and H. Seyalioglu, "Fully secure accountable-authority identity based encryption," in *The 14th International Conference on Practice and Theory in Public Key Cryptography*, pp. 296–316, 2011.

[28] A. Sahai and B. Waters, "Fuzzy identity-based encryption," in *The 24th Annual International Conference on Theory and Applications of Cryptographic Techniques*, pp. 457–473, 2005.

[29] S. Wang, K.Guo, and Y. Zhang, "Traceable ciphertext-policy attribute-based encryption scheme with attribute level user revocation for cloud storage," *PLOS ONE*, vol. 13, no. 9, pp. 1–23, Sep. 2018.

[30] Y. Wang, K. Chen, Y. Long, and Z. Liu, "Accountable authority key policy attribute-based encryption," *Science China-Information Sciences*, vol. 55, no. 7, pp. 1631–1638, 2012.

[31] G. Yu, X. Ma, Z. Cao, G. Zeng, and W. Han, "Accountable CP-ABE with public verifiability: How to effectively protect the outsourced data in cloud," *International Journal of Foundations of Computer Science*, vol. 28, no. 6, pp. 705–723, 2017.

[32] G. Yu, X. Ma, Z. Cao, W. Zhu, and J. Zeng, "Accountable multi-authority ciphertext-policy attribute-based encryption without key escrow and key abuse," in *International Symposium on Cyberspace Safety and Security*, pp. 337–351, Oct. 2017.

[33] G. Yu, Y. Wang, Z. Cao, J. Lin, and X. Wang, "Traceable and undeniable ciphertext-policy attribute-based encryption for cloud storage service," *International Journal of Distributed Sensor Networks*, vol. 15, no. 4, pp. 1–10, 2019.

[34] K. Zhang, H. Li, J. Ma, and X. Liu, "Efficient large-universe multi-authority ciphertext-policy attribute-based encryption with white-box traceability," *Science China-Information Sciences*, vol. 61, no. 3, pp. 1–14, Mar. 2018.

[35] X. Zhang, C. Jin, C. Li, Z. Wen, Q. Shen, Y. Fang, and Z. Wu, "Ciphertext-policy attribute-based encryption with user and authority accountability," in *The 11th International Conference Security and Privacy in Communication Networks*, pp. 500–518, 2015.

[36] Y. Zhang, J. Li, D. Zheng, X. Chen, and H. Li, "Towards privacy protection and malicious behavior traceability in smart health," *Personal and Ubiquitous Computing*, vol. 21, no. 5, pp. 815–830, 2017.

[37] Q. Zhao, G. Wu, H. Ma, Y. Zhang, and H. Wang, "Black-box and public traceability in multi-authority attribute based encryption," *Chinese Journal of Electronics*, vol. 29, no. 1, pp. 106–113, Apr. 2020.

[38] Z. Zhao, J. Lai, W. Susilo, B. Wang, Y. Hu, and F. Guo, "Efficient construction for full black-box accountable authority identity based encryption," *IEEE Access*, vol. 7, pp. 25936–25947, 2019.

Biography

Leyou Zhang received the M.S. and Ph.D. degrees from Xidian University, in 2002 and 2009, respectively. He is currently a Professor with Xidian University. His current research interests include cryptography, network security, cloud security, and computer security.

Chao Song received the B.S. degree in engineering from Guizhou University, in China, in 2018. He is currently pursuing the M.S. degree in applied information security with Xidian University, China. His current interests include cryptography and cloud security.

Yi Mu received the Ph.D. degree from Australian National University, in 1994. He is currently a Professor with Fujian Normal University (FJNU). Prior to his appointment at FJNU, he is also a Full Professor with the School of Computing and Information Technology, University of Wollongong. His current research interests include cryptography, network security, access control, and computer security. He also previously worked in the areas of quantum cryptography, quantum computers, atomic computations, and quantum optics.

Some Properties and Privacy Measurement of 0/1-Encoding

Ya-Ting Duan[1,2], Yan-Ping Li[1,2], Lai-Feng Lu[1], and Kai Zhang[1]
(Corresponding author: Yan-Ping Li)
School of Mathematics and Statistics, Shaanxi Normal University[1]
Xi'an 710119, China
Email: lyp@snnu.edu.cn
State key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications[2]
(Received Feb. 5, 2021; Revised and Accepted June 6, 2021; First Online Dec. 18, 2021)

Abstract

The 0/1-encoding is a new encoding method, mainly used to compare the numerical size of two positive integers without giving the specific integers. Its main idea is to reduce the problem of comparing two integers to the problem to find whether two sets have intersections. In this paper, we first analyze and discuss the inherent properties of the 0/1-encoding method by three theorems. Then, if the 0/1-encoding sets are used directly, it will make the adversary get 0/1-encoding results easily, and the adversary has a greater probability of recovering the positive integers being compared. Finally, we theoretically prove the above findings and depict the degree of privacy leakage compared to positive integers when the adversary obtains different 0/1-encoding results.

Keywords: 0/1-Encoding Method; Greater Than (GT) Problem; Millionaire Problem; Privacy Leakage; Set Intersection Problem

1 Introduction

The Millionaire problem is a well-known problem in cryptography [18], which is similar to the greater than (GT) problem [10], namely, to determine which of two integers is larger without leaking any information about integer values being compared. Since Yao introduced this problem and gave a solution, many scheme [1,6,8,10,15] have been put forward to solve this problem using different methods. In particular, Lin *et al.* [10] proposed a special encoding method, namely 0/1-encoding, which transformed the greater than problem into the set intersection problem. Because of the simplicity of the 0/1-encoding method, more and more researchers pay attention to it and it is widely applied in many fields.

References [2,3,5,7,11,12,14,16,19,20] introduced the 0/1-encoding into their schemes to solve problems in different fields. The schemes [7,11,14] introduced the 0/1-encoding into the time-limited signature. Firstly, the key expiration time T_1 was embedded into the user's private key, and then T_1 and the signature time T_2 were encoded according to the 1-encoding and 0-encoding respectively. If there was a common element in these two encoding sets, then the signature was valid and can be verified. The specific problems in [2,3,5,12,16,19,20] were also solved by the 0/1-encoding method. For example, Shishido *et al.* [16] proposed a test scheme to judge whether an integer d belongs to a range of $[a,b]$. This scheme first encoded the left and right endpoints a and b with 1-encoding [4] and 0-encoding, respectively, and then detected whether the prefix string set [16] of integer d has common elements with the encoding sets of two endpoints. If they had a common element, then d can be judged to be out of the range $[a,b]$.

In this paper, we find that if the 0/1-encoding sets are directly used without any processing, it may lead to the privacy leakage of the integer values being compared, which obviously contradicts the original intention of the 0/1-encoding method obviously. For example, some attribute-based encryption schemes [13,17] used the 0/1-encoding method to compare whether the numerical attributes (height, weight, or age) of the data owners and the data users are matched or not. These numerical attributes often are the personal privacy and the user does not want to expose their privacy, so they make indirect comparison using the 0/1-encoding method. If all 0/1-encoding results are stored in the cloud server, and any entity who obtains the encoding results can recover the exact values of these numerical attributes according to the original encoding sets, which is contrary to the original designed intention of the 0/1-encoding method. Therefore, this paper analyzes and discusses the properties and defects of the 0/1-encoding method, and points out the precautions for its use. Here, we summarize the contributions of this paper as follows:

- We analyze and discuss the inherent properties of the

0/1-encoding method by three theorems, and further extend the 0/1-encoding method to indirectly judge the relations (" $>$ ", " $\leq$ " and " $=$ ") of two positive integers.

- We find that if the 0/1-encoding is used directly, it will make the adversary get 0/1-encoding results easily and the adversary has a greater probability to recover the positive integers being compared, and prove the findings by a theorem.
- We depict the degree of privacy leakage of the compared positive integers when the adversary obtains different 0/1-encoding results by two theorems and some examples.

The remainder of this paper is organized as follows. Section 2 reviews some basic knowledge which might be used. Section 3 gives three theorems to introduce properties of the 0/1-encoding method. Section 4 describes the correlation between the exposure of the 0/1 encoding sets and the privacy leakage of the compared integers via three theorems and specific examples. Section 5 gives some suggestions for 0/1-encoding to protect the privacy of integers being compared.

2 Preliminaries

Definition 1. *(see [10]) Let $x = x_n x_{n-1}...x_1 \in GF_2^n$ be a n-bit binary string, then the 0/1-encoding set of x are defined as sets S_x^0 and S_x^1, which are shown below:*

$$\begin{aligned} S_x^0 &= \{x_n x_{n-1}...x_{i+1}1 | x_i = 0, 1 \leq i \leq n\}, \\ S_x^1 &= \{x_n x_{n-1}...x_i | x_i = 1, 1 \leq i \leq n\}. \end{aligned}$$

Both S_x^0 and S_x^1 have at most n elements.

To compare which of two integers x and y is greater by the 0/1-encoding method, we first encode them as binary strings of the same length (and if the length is not equal, the top should be added with 0 to the same length), then determine whether there is a non-empty intersection between the 1-encoding set S_x^1 of x and the 0-encoding set S_y^0 of y (or between the 0-encoding set S_x^0 of x and the 1-encoding set S_y^1 of y). If $S_x^1 \bigcap S_y^0 \neq \varnothing$, we can determine that $x > y$. Otherwise, $x \leq y$. We illustrate it with the following example.

Example 1. *Let $x = 45, y = 40$ and their binary strings be 101101 and 101000 respectively. Then the 0/1-encoding sets of x and y are*

$$\begin{aligned} S_x^0 &= \{10111, 11\}, S_x^1 = \{101101, 1011, 101, 1\}. \\ S_y^0 &= \{101001, 10101, 1011, 11\}, S_x^1 = \{101, 1\}. \end{aligned}$$

Since $S_x^1 \bigcap S_y^0 = \{1011\} \neq \varnothing$, we can infer that $x > y$. Of course, we also can get $y \leq x$ by $S_x^0 \bigcap S_y^1 = \varnothing$. By the 0/1-encoding, we can get a conclusion that x is greater than y.

3 Main Properties

In this section, three theorems are given to illustrate how to determine the relations (" $>$ ", " $\leq$ " and " $=$ ") of two positive integers by the 0/1-encoding method, which indicates that the 0/1-encoding can indirectly judge the above relations (" $>$ ", " $\leq$ " and " $=$ ") of two integers. First, we give some notations and their meanings in Table 1 that might be used later.

Theorem 1. Let x and y are any two positive integers. Then $x > y$ if and only if S_x^1 and S_y^0 have only a common element.

Proof. Let $x = x_n x_{n-1}...x_1 \in GF_2^n, y = y_n y_{n-1}...y_1 \in GF_2^n$.

Prove sufficiency. Suppose there is an element $t = t_n t_{n-1}...t_i \in S_x^1 \bigcap S_y^0$ with $t_i = 1, i \in [1, n]$. Since $t \in S_x^1$, there must exsit $x_n x_{n-1}...x_i = t_n t_{n-1}...t_i$. And due to $t \in S_y^0$, we can get that $y_n y_{n-1}...y_i = t_n t_{n-1}...\bar{t_i}$, namely

$$\begin{cases} x_j = y_j, & j \in [i+1, n] \\ x_i = 0, y_i = 0, & i = j \end{cases}.$$

Therefore, we can infer that $x > y$.

Prove necessity. We first prove that S_x^1 and S_y^0 have a common element, that is, prove the existence of the common element. If $x > y$, then there must be an integer $i \in [1, n]$, such that

$$\begin{cases} x_j = y_j, & j \in [i+1, n] \\ x_i = 0, y_i = 0, & i = j \end{cases} \quad (1)$$

where i must be the first value of i that satisfies the above condition (1). From the above conditions (1), we can know $x_n x_{n-1}...x_{i+1}x_i = y_n y_{n-1}...y_{i+1}1$. According to the 1-encoding set S_x^1 of x and the 0-encoding set S_y^0 of y, it is easy to get $x_n x_{n-1}...x_{i+1}x_i \in S_x^1$ and $y_n y_{n-1}...y_{i+1}1 \in S_y^0$. Hence, $x_n x_{n-1}...x_{i+1}x_i = y_n y_{n-1}...y_{i+1}1 \in S_x^1 \bigcap S_y^0$, namely, S_x^1 and S_y^0 have a common element.

Next, we will prove that S_x^1 and S_y^0 have only a common element, i.e., the uniqueness of the common element. Assume there is another common element $t \in S_x^1 \bigcap S_y^0$, then t must be represented as follows:

$$t = x_n...x_{i+1}x_i x_{i-1}...x_{j+1}x_j \text{ (or } y_n...y_{i+1}1y_{i-1}...y_{j+1}1).$$

Obviously, the corresponding values of the above two binary strings are equal, namely, $x_k = y_k, k \in [j, n]$, where $y_i = y_j = 1$. Since $t \in S_y^0$, there be $y_n y_{n-1}...y_j = y_n y_{n-1}...y_{i+1}1y_{i-1}...y_{j+1}0$, and then $y_n y_{n-1}...y_{i+1}y_i = y_n y_{n-1}...y_{i+1}1$. And due to $y_n y_{n-1}...y_{i+1}1 \in S_y^0$, we can infer that $y_n y_{n-1}...y_{i+1}y_i = y_n y_{n-1}...y_{i+1}0$. This is a contradiction obviously. Therefore, S_x^1 and S_y^0 have only a common element.

Table 1: Notations

notations	meanings	notations	meanings
$[1,n]$	all integers between 1 and n	GF_2	Galois field
$\bar{t}$	complement of t in GF_2	$\|S\|$	the cardinality of the set S
C_n^m	the combinations number $\frac{n!}{m!(n-m)!}$	$\|\|$	cascading symbol of binary strings

□

Remark 1. From the uniqueness proof process of Theorem 1, we know that their encoding sets S_x^1 and S_y^0 (or S_x^0 and S_y^1) do not have two or more common elements for any two positive integers x and y. Therefore, the following $S_x^1 \bigcap S_y^0 \neq \varnothing$ (or $S_x^0 \bigcap S_y^1 \neq \varnothing$) indicates that S_x^1 and S_y^0 (or S_x^0 and S_y^1) have only a common element.

Theorem 2. Let x and y be any two positive integers. Then $x \leq y \Leftrightarrow S_x^1 \bigcap S_y^0 = \varnothing$.

Proof. We use apagoge to prove it. Given $S_x^1 \bigcap S_y^0 = \varnothing$, if $x > y$, then S_x^1 and S_y^0 have only a common element. This is contradictory with $S_x^1 \bigcap S_y^0 = \varnothing$, so $x \leq y$. On the contrary, let $x \leq y$, if $S_x^1 \bigcap S_y^0 \neq \varnothing$, namely, S_x^1 and S_y^0 have only a common element, then $x > y$ according to Theorem 1. It contradicts with $x \leq y$, so $S_x^1 \bigcap S_y^0 = \varnothing$. □

Theorem 3. Let x and y be any two positive integers. The following three conditions are equivalent*:*

1) $x = y$*;*

2) $S_x^1 \bigcap S_y^0 = \varnothing$ and $S_x^0 \bigcap S_y^1 = \varnothing$*;*

3) $S_x^1 \bigcap S_y^0 = \varnothing$ and $S_x^1 \bigcap S_{y-1}^0 \neq \varnothing$.

Proof.

(1) ⇒ (2) $x = y \Leftrightarrow x \leq y$ **and** $y \leq x$**.** According to Theorem 2, we can get that

$$x \leq y \Leftrightarrow S_x^1 \bigcap S_y^0 = \varnothing, y \leq x \Leftrightarrow S_y^1 \bigcap S_x^0 = \varnothing.$$

Therefore, $S_x^1 \bigcap S_y^0 = \varnothing$ and $S_x^0 \bigcap S_y^1 = \varnothing$.

(2) ⇒ (3)**.** We just need to prove $S_x^1 \bigcap S_{y-1}^0 \neq \varnothing$. From Theorem 2, we know that

$$S_x^0 \bigcap S_y^1 = \varnothing \Rightarrow y \leq x.$$

Since $y - 1 < y$, we can get $y - 1 < x$. It is known by Theorem 1 that

$$x > y - 1 \Leftrightarrow S_x^1 \text{ and } S_{y-1}^0 \text{ have only a common element.}$$

Therefore, $S_x^1 \bigcap S_{y-1}^0 \neq \varnothing$.

(3) ⇒ (1)**.** Suppose that $S_x^1 \bigcap S_y^0 = \varnothing$ and $S_x^1 \bigcap S_{y-1}^0 \neq \varnothing$. According to Theorem 1 and Theorem 2

$$S_x^1 \bigcap S_y^0 = \varnothing \Rightarrow x \leq y,\ S_x^1 \bigcap S_{y-1}^0 \neq \varnothing \Rightarrow x > y - 1.$$

That is, $y-1 < x \leq y$. And due to x be a positive integer, then $x = y$. □

4 Correlation between the 0/1-Encoding Sets and the Privacy of Compared Integer

If the 0/1-encoding sets are not processed and directly submitted to the third party or both parties exchange the 0/1-encoding sets to find whether there is a non-empty intersection, which may lead to the privacy leakage of the integer values being compared. This contradicts with the original designed intention of this encoding method. The following will illustrate this fact via three theorems and concrete examples.

Theorem 4. If the length(n-bit) of binary string of a positive integer x is known, and

$$|S_x^1| = n_1 \ (\text{or } |S_x^0| = n_2),$$

then we have the probability of $\frac{1}{C_n^{n_1}}$ (or $\frac{1}{C_n^{n-n_2}}$) to recover x, where $n = n_1 + n_2$.

Proof. Let $x = t_n t_{n-1} ... t_1 \in GF_2^n$. If $|S_x^1| = n_1$, then there are n_1 values of t_i ($i \in [1,n]$) which are equal to 1. And there are totally $C_n^{n_1}$ positions that locate $t_i = 1$, thus there is the probability of $\frac{1}{C_n^{n_1}}$ to get x. Similarly, if we know $|S_x^0| = n_2$, we can also prove that there is the probability of $\frac{1}{C_n^{n-n_2}}$ to recover x value. □

Example 2. *Suppose that an adversary knows the number of elements in a 0/1-encoding set, namely,* $|S_x^1| = 3$ *(or* $|S_x^0| = 3$*), then the adversary has the probability of* $\frac{1}{C_6^3} = \frac{1}{20}$ *to get* x*. See Table 2.*

Obviously, there are 20 possible x *values. However, since* x *is usually associated with a specific attribute (height, weight, or age), the adversary can further determine* x *based on other information. For example,* x *represents the age of the breast cancer patient in the example, then the range of* x *can be determined to [40,56], and the possible values of* x *are 41, 42, 44, 49, 50, 52, 56. Therefore, the adversary has the probability of* $\frac{1}{7}$ *to get* x*.*

Next, we first discuss the rule of binary strings of the 0/1-encoding sets for a certain integer x *by Theorem 5, and then summarize how to recover the integer* x *from the 0/1-encoding sets and prove the fact with Theorem 6.*

Theorem 5. Assume that $x = x_k^{(i)} = t_k t_{k-1} ... t_1 \in GF_2^k$, then the 0/1-encoding set of $x_k^{(i)}$ can be represented by

Table 2: All the possible values of x

x	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}
binary	111000	110100	110010	110001	101100	101010	101001	100110	100101	100011
decimal	56	52	50	49	44	42	41	38	37	35
x	x_{11}	x_{12}	x_{13}	x_{14}	x_{15}	x_{16}	x_{17}	x_{18}	x_{19}	x_{20}
binary	011100	011010	011001	010110	010101	010011	001110	001101	001011	000111
decimal	28	26	25	22	21	19	14	13	11	7

the 0/1 encoding set of $x_{k-1}^{(j)}$ as follows.

$$S^0_{x_k^{(i)}} = \begin{cases} \left\{1,0||c|c \in S^0_{x_{k-1}^{(j)}}\right\}, & t_k = 0 \\ \left\{1||c|c \in S^0_{x_{k-1}^{(j)}}\right\}, & t_k = 1 \end{cases},$$

$$S^1_{x_k^{(i)}} = \begin{cases} \left\{0||c|c \in S^1_{x_{k-1}^{(j)}}\right\}, & t_k = 0 \\ \left\{1,1||c|c \in S^1_{x_{k-1}^{(j)}}\right\}, & t_k = 1 \end{cases},$$

where $j = \begin{cases} i, & 1 \le i \le 2^{k-1} \\ i - 2^{k-1}, & 2^{k-1} < i \le 2^k \end{cases}$.

Proof. We will use the mathematical induction to prove it.

Let $n = 1, x = x_1^{(i)} = t_1 \in GF_2$, then there are two cases:

$$x_1^{(1)} = 0, S^0_{x_1^{(1)}} = \{1\}, S^1_{x_1^{(1)}} = \varnothing;$$
$$x_1^{(2)} = 1, S^0_{x_1^{(2)}} = \varnothing, S^1_{x_1^{(2)}} = \{1\}.$$

Let $n = 2, x = x_2^{(i)}) = t_2t_1 \in GF_2^2$, then there are four cases:

$$x_2^{(1)} = 00, S^0_{x_2^{(1)}} = \{1,01\}, S^1_{x_2^{(1)}} = \varnothing;$$
$$x_2^{(2)} = 01, S^0_{x_2^{(2)}} = \{1\}, S^1_{x_2^{(2)}} = \{01\};$$
$$x_2^{(3)} = 10, S^0_{x_2^{(3)}} = \{11\}, S^1_{x_2^{(3)}} = \{1\};$$
$$x_2^{(4)} = 11, S^0_{x_2^{(4)}} = \varnothing, S^1_{x_2^{(4)}} = \{1,11\}.$$

Let $n = 3, x = x_3^{(i)}) = t_3t_2t_1 \in GF_2^3$, then there are eight cases:

$$x_3^{(1)} = 000, S^0_{x_3^{(1)}} = \{1,01,001\}, S^1_{x_3^{(1)}} = \varnothing;$$
$$x_3^{(2)} = 001, S^0_{x_3^{(2)}} = \{1,01\}, S^1_{x_3^{(2)}} = \{001\};$$
$$x_3^{(3)} = 010, S^0_{x_3^{(3)}} = \{011,1\}, S^1_{x_3^{(3)}} = \{01\};$$
$$x_3^{(4)} = 011, S^0_{x_3^{(4)}} = \{1\}, S^1_{x_3^{(4)}} = \{01,011\};$$
$$x_3^{(5)} = 100, S^0_{x_3^{(5)}} = \{101,11\}, S^1_{x_3^{(5)}} = \{1\};$$
$$x_3^{(6)} = 101, S^0_{x_3^{(6)}} = \{11\}, S^1_{x_3^{(6)}} = \{1,101\};$$
$$x_3^{(7)} = 110, S^0_{x_3^{(7)}} = \{111\}, S^1_{x_3^{(7)}} = \{1,11\};$$
$$x_3^{(8)} = 111, S^0_{x_3^{(8)}} = \varnothing, S^1_{x_3^{(8)}} = \{1,11,111\}.$$

From above, we can see that

$$S^0_{x_2^{(i)}} = \begin{cases} \left\{1,0||c|c \in S^0_{x_1^{(j)}}\right\}, & t_2 = 0 \\ \left\{1||c|c \in S^0_{x_1^{(j)}}\right\}, & t_2 = 1 \end{cases},$$

$$S^1_{x_2^{(i)}} = \begin{cases} \left\{0||c|c \in S^1_{x_1^{(j)}}\right\}, & t_2 = 0 \\ \left\{1,1||c|c \in S^1_{x_1^{(j)}}\right\}, & t_2 = 1 \end{cases},$$

where $j = \begin{cases} i, & 1 \le i \le 2 \\ i - 2, & 2 < i \le 2^2 \end{cases}$.

$$S^0_{x_3^{(i)}} = \begin{cases} \left\{1,0||c|c \in S^0_{x_2^{(j)}}\right\}, & t_3 = 0 \\ \left\{1||c|c \in S^0_{x_2^{(j)}}\right\}, & t_3 = 1 \end{cases},$$

$$S^1_{x_3^{(i)}} = \begin{cases} \left\{0||c|c \in S^1_{x_2^{(j)}}\right\}, & t_3 = 0 \\ \left\{1,1||c|c \in S^1_{x_2^{(j)}}\right\}, & t_3 = 1 \end{cases},$$

where $j = \begin{cases} i, & 1 \le i \le 2^2 \\ i - 2^2, & 2^2 < i \le 2^3 \end{cases}$. So when $n = 2$ and $n = 3$, the conclusion holds.

Now suppose $n = k - 1$, the conclusion also holds, namely

$$S^0_{x_{k-1}^{(i)}} = \begin{cases} \left\{1,0||c|c \in S^0_{x_{k-2}^{(j)}}\right\}, & t_{k-1} = 0 \\ \left\{1||c|c \in S^0_{x_{k-2}^{(j)}}\right\}, & t_{k-1} = 1 \end{cases},$$

$$S^1_{x_{k-1}^{(i)}} = \begin{cases} \left\{0||c|c \in S^1_{x_{k-2}^{(j)}}\right\}, & t_{k-1} = 0 \\ \left\{1,1||c|c \in S^1_{x_{k-2}^{(j)}}\right\}, & t_{k-1} = 1 \end{cases},$$

where $j = \begin{cases} i, & 1 \le i \le 2^{k-2} \\ i - 2^{k-2}, & 2^{k-2} < i \le 2^{k-1} \end{cases}$. Then when $n = k$, namely, $x = x_k^{(i)} = t_kt_{k-1}...t_1 \in GF_2^k$, there are two cases:

1) $t_k = 0$.

a. Let $t_{k-1}=0$, that is, $x_k^{(i)}=00t_{k-2}...t_1$, we can get $S^0_{x_k^{(i)}}=\left\{1,01,0||0||c|c\in S^0_{x_{k-2}^{(j)}}\right\}$ and $S^1_{x_k^{(i)}}=\left\{0||0||c|c\in S^1_{x_{k-2}^{(j)}}\right\}$, where $1\le i=j\le 2^{k-2}$.

b. Let $t_{k-1}=1$, namely, $x_k^{(i)}=01t_{k-2}...t_1$, we can get $S^0_{x_k^{(i)}}=\left\{1,0||0||c|c\in S^0_{x_{k-2}^{(j)}}\right\}$ and $S^1_{x_k^{(i)}}=\left\{01,0||0||c|c\in S^1_{x_{k-2}^{(j)}}\right\}$, where $j=i-2^{k-2}, 2^{k-2}\le i\le 2^{k-1}$.

So when $t_k=0$, we can get $S^0_{x_k^{(i)}}=\left\{1,0||c|c\in S^0_{x_{k-1}^{(j)}}\right\}, S^1_{x_k^{(i)}}=\left\{0||c|c\in S^1_{x_{k-1}^{(j)}}\right\}$, where $1\le i=j\le 2^{k-1}$.

2) $t_k=1$.

a. Let $t_{k-1}=0$, that is to say, $x_k^{(i)}=10t_{k-2}...t_1$, we can get $S^0_{x_k^{(i)}}=\left\{11,1||0||c|c\in S^0_{x_{k-2}^{(j)}}\right\}$ and $S^1_{x_k^{(i)}}=\left\{1,1||0||c|c\in S^1_{x_{k-2}^{(j)}}\right\}$, where $j=i-2^{k-1}, 2^{k-1}<i\le 3\times 2^{k-2}$.

b. Let $t_{k-1}=1$, that means $x_k^{(i)}=11t_{k-2}...t_1$, we can get $S^0_{x_k^{(i)}}=\left\{1||1||c|c\in S^0_{x_{k-2}^{(j)}}\right\}$ and $S^1_{x_k^{(i)}}=\left\{1,11,1||1||c|c\in S^1_{x_{k-2}^{(j)}}\right\}$, where $j=i-3\times 2^{k-2}, 3\times 2^{k-2}<i\le 2^k$.

So when $t_k=1$, $S^0_{x_k^{(i)}}=\left\{1||c|c\in S^0_{x_{k-1}^{(j)}}\right\}$ and $S^1_{x_k^{(i)}}=\left\{1,1||c|c\in S^1_{x_{k-1}^{(j)}}\right\}$ hold, where $j=i-2^{k-1}, 2^{k-1}<i\le 2^k$.

Therefore, when $n=k$, we can get

$$S^0_{x_k^{(i)}}=\begin{cases}\left\{1,0||c|c\in S^0_{x_{k-1}^{(j)}}\right\}, & t_k=0\\ \left\{1||c|c\in S^0_{x_{k-1}^{(j)}}\right\}, & t_k=1\end{cases},$$

$$S^1_{x_k^{(i)}}=\begin{cases}\left\{0||c|c\in S^1_{x_{k-1}^{(j)}}\right\}, & t_k=0\\ \left\{1,1||c|c\in S^1_{x_{k-1}^{(j)}}\right\}, & t_k=1\end{cases},$$

where $j=\begin{cases}i, & 1\le i\le 2^{k-1}\\ i-2^{k-1}, & 2^{k-1}<i\le 2^k\end{cases}$.

To sum up, the conclusion of Theorem 5 is true. □

Remark 2. When $S^1_{x_k^{(i)}}=\varnothing$ (or $S^0_{x_k^{(i)}}=\varnothing$), $\left\{1||c|c\in S^1_{x_k^{(i)}} \text{ or } S^0_{x_k^{(i)}}\right\}=\varnothing$.

Theorem 6. Let x be an any positive integer. Given the 0/1-encoding sets S_x^0 and S_x^1, the value of x must be recoverable.

Proof. Given a positive integer x, it must be equal to some $x_n^{(i)}$. Thus we can recover x according to Theorem 5 theoretically. The proof process of Theorem 5 shows that we can recover the value of $x_n^{(i)}$ when given the corresponding 0/1-encoding sets $S^0_{x_n^{(i)}}$ and $S^1_{x_n^{(i)}}$, where $n=1, 2, 3$. When $n=4$, given the 0/1-encoding sets $S^0_{x_4^{(i)}}$ and $S^1_{x_4^{(i)}}$ of $x_4^{(i)}$, they must be represented by the 0/1-encoding sets $S^0_{x_3^{(j)}}$ and $S^1_{x_3^{(j)}}$ of $x_3^{(j)}$, where i and j satisfy Theorem 5. Specifically, we just need to figure out the corresponding $x_3^{(j)}$. Assume $x_3^{(j)}=t_3t_2t_1$, if the 0/1-encoding sets of $x_4^{(i)}$ and $x_3^{(j)}$ satisfy the first case of Theorem 5, then $x_4^{(i)}=0t_3t_2t_1$. Otherwise, $x_4^{(i)}=1t_3t_2t_1$. In this way, we can certainly recover the value of $x_n^{(i)}$ when $n\le k-1$.

When $n=k$, given the 0/1-encoding sets $S^0_{x_k^{(i)}}$ and $S^1_{x_k^{(i)}}$, we can definitely find the corresponding $x_{k-1}^{(j)}$, where the 0-encoding sets $S^0_{x_{k-1}^{(j)}}$ and $S^0_{x_k^{(i)}}$ (and the 1-encoding sets $S^1_{x_{k-1}^{(j)}}$ and $S^1_{x_k^{(i)}}$) satisfy Theorem 5. Suppose $x_{k-1}^{(j)}=t_{k-1}...t_2t_1$, we can get $x_k^{(i)}=0t_{k-1}...t_2t_1$ or $x_k^{(i)}=1t_{k-1}...t_2t_1$ according to Theorem 5.

To sum up, if S_x^0 and S_x^1 of x are given, we can definitely recover the value of x theoretically. □

Remark 3. In the above proof, we can recover the value $x=x_n^{(i)}$ by finding the corresponding $x_{n-1}^{(j)}$, and recover the value of $x_{n-1}^{(j)}$ by the corresponding $x_{n-2}^{(k)}$, and so on. We can finally get the value of x using the recursion method. However, it is a little bit tedious, Theorem 5 just claims that x can be recovered theoretically.

In the following, we will give a specific method to recover the value of x.

Proof. Assume $x=t_nt_{n-1}...t_1\in GF_2^n$ and t be the longest binary string of the 0/1-encoding sets of x, then there must be two cases:

1) If $t_1=1$, then $t=t_nt_{n-1}...t_1\in S_x^1$;

2) If $t_1=0$, then $t=t_nt_{n-1}...\bar{t_1}\in S_x^0$;

Given the 0/1-encoding sets S_x^0 and S_x^1 of x, it is easy to find the longest binary string $t=y_ny_{n-1}...y_1$ of two sets, and then we can determine the relationship x and t according to the set which t belongs to. Finally, we can recover x, i.e.,

$$\begin{cases}x=t=y_ny_{n-1}...y_1, & t\in S_x^1\\ x=y_ny_{n-1}...\bar{y_1}, & t\in S_x^0\end{cases}.$$

From above, we can see that the value of x depends on the set which the longest binary string t belongs to, and

it mainly depends on the value of t_1 ($t_1 \in GF_2$). Given an any positive integer x, it must be

$$t_1 = \begin{cases} 0, & x = 2n \\ 1, & x = 2n-1 \end{cases}, n \in N^*.$$

Since $Pr(x = 2n) = Pr(x = 2n-1) = \frac{1}{2}$, so $Pr(t_1 = 1) = Pr(t_1 = 0) = \frac{1}{2}$, namely $Pr(t \in S_x^1) = Pr(t \in S_x^0) = \frac{1}{2}$. Therefore, if an adversary only has a set S_x^0 or S_x^1 of x, it also has a probability of $\frac{1}{2}$ to recover the value of x. □

Example 3. *The adversary captures two encoding sets of x (and it maybe do nor know which set of S_1 and S_2 is the 0-encoding set or 1-encoding set),*

$$\begin{aligned} S_1 &= \{10110101, 1011011, 10111, 11\}, \\ S_2 &= \{101101, 1011, 101, 1\}. \end{aligned}$$

The adversary can first find the longest binary string $t = 10110101$ and $t \in S_1$. By observing the relationship of these elements in S_1, it can be found that S_1 is 0-encoding set (because short codes must be prefixed to long codes in the 1-encoding set). Hence, $t \in S_1 = S_x^0$ and $x = 10110100 = 180$. To sum up, even if the adversary captures part of the 0/1-encoding sets, there still exists a certain probability that the adversary will recover the integer values being compared.

5 Conclusions

In this paper, three theorems are given to illustrate how to determine the relations (" $>$ ", " $\leq$ " and " $=$ ") of two positive integers by the 0/1-encoding method. Then, another three theorems and related examples show that if the 0/1-encoding results are not blindly preprocessed, it is easy to leak the integer values being compared, which obviously contradicts the original designed intention of the encoding method. Since the privacy of integer values in many fields, when using the 0/1-encoding method for numerical comparison, the 0/1-encoding sets should be properly encrypted or blinded, or the intersection of two 0/1-encoding sets should be calculated confidentially [9] to avoid privacy leakage.

Acknowledgments

This work was supported by National Natural Science Foundation of China (61802243), the Key R&D program in industry field of Shaanxi Province (2019GY-013) and the basic science research program of Shaanxi Province (2019JQ273,2020JM288), and Open Foundation of State key Laboratory of Networking and Switching Technology (Beijing University of Posts and Telecommunications) (SKLNST-2020-1-03). The authors gratefully acknowledge the anonymous reviewers for their valuable comments.

References

[1] I. F. Blake and V. Kolesnikov, "Strong conditional oblivious transfer and computing on intervals," in *International Conference on the Theory and Application of Cryptology and Information Security*, pp. 515–529, 2004.

[2] H. Dai, T. Wei, Y. Huang, J. Xu, and G. Yang, "Random secure comparator selection based privacy-preserving max/min query processing in two-tiered sensor networks," *Journal of Sensors*, vol. 2016, pp. 1–13, 2015.

[3] J. N. Doctor, J. Vaidya, X. Jiang, W. Shuang, and D. Meeker, "Efficient determination of equivalence for encrypted data," *Computers and Security*, vol. 97, 2018. (`https://doi.org/10.1016/j.cose.2020.101939`)

[4] Y. Dou, H. C. B. Chan, and M. H. Au, "Order-hiding range query over encrypted data without search pattern leakage," *The Computer Journal*, vol. 61, no. 12, pp. 1806–1824, 2018.

[5] A. Dupin, J. M. Robert, and C. Bidan, "Location-proof system based on secure multi-party computations," *IACR Cryptology ePrint Archive*, vol. 2018, pp. 525, 2018.

[6] I. Ioannidis and A. Grama, "An efficient protocol for Yao's millionaires' problem," in *Proceedings of the 36th Annual Hawaii International Conference on System Sciences*, 2003. DOI: 10.1109/HICSS.2003.1174464.

[7] I. Karnil, O. Olakanmi, and S. O. Ogundoyin, "A secure and privacy-preserving lightweight authentication protocol for wireless communications," *Information Systems Security*, vol. 26, no. 4-6, pp. 287–304, 2017.

[8] S. D. Li, Y. Q. Dai, and Q. Y. You, "An efficient solution to Yao's millionaires' problem (in chinese)," *Acta Electronica Sinica*, vol. 33, no. 5, pp. 769–773, 2005.

[9] S. D. Li, S. F. Zho, Y. M. Guo, J. W. Dou, and D. S. Wang, "Secure set computing in cloud environment (in chinese)," *Journal of Software*, 2016. DOI:10.13328/j.cnki.jos.004996.

[10] H. Y. Lin and W. G. Tzeng, "An efficient solution to the millionaires' problem based on homomorphic encryption," *IACR Cryptology ePrint Archive*, vol. 2005, pp. 43–43, 2005.

[11] J. K. Liu, C. Chu, S. S. M. Chow, X. Huang, M. H. Au, and J. Zhou, "Time-bound anonymous authentication for roaming networks," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 1, pp. 178–189, 2014.

[12] M. S. I. Mamun and A. Miyaji, "Secure VANET applications with a refined group signature," in *Twelfth Annual International Conference on Privacy, Security and Trust*, 2014. DOI: 10.1109/PST.2014.6890940.

[13] Y. Miao, J. Ma, X. Liu, X. Li, Z. Liu, and H. Li, "Practical attribute-based multi-keyword

search scheme in mobile crowdsourcing," *IEEE Internet of Things Journal*, vol. 5, no. 4, pp. 1–1, 2017.
[14] O. S. Oyinlola and A. S. Oladele, "Edas: Efficient data aggregation scheme for internet of things," *Journal of Applied Security Research*, vol. 13, no. 3, pp. 347–375, 2018.
[15] B. Schoenmakers and P. Tuyls, "Practical two-party computation based on the conditional gate," in *International Conference on the Theory and Application of Cryptology and Information Security*, vol. 3329, pp. 119–136, 2004.
[16] K. Shishido and A. Miyaji, "Secure online-efficient interval test based on empty-set check," in *The 14th Asia Joint Conference on Information Security (AsiaJCIS'19)*, 2019. DOI: 10.1109/AsiaJCIS.2019.000-5.
[17] K. Xue, J. Hong, Y. Xue, D. S. L. Wei, N. Yu, and P. Hong, "Cabe: A new comparable attribute-based encryption construction with 0-encoding and 1-encoding," *IEEE Transactions on Computers*, vol. 66, no. 9, pp. 1491–1503, 2017.
[18] A. C. Yao, "Protocols for secure computations," in *Proceedings of the 23rd Annual IEEE Symposium on Foundations of Computer Science*, 1982. DOI: 10.1109/SFCS.1982.38.
[19] L. Zhang, J. Song, and J. Pan, "A privacy-preserving and secure framework for opportunistic routing in dtns," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 9, pp. 7684–7697, 2016.
[20] B. Zhao, S. Tang, X. Liu, X. Zhang, and W. N. Chen, "Ironm: Privacy-preserving reliability estimation of heterogeneous data for mobile crowdsensing," *IEEE Internet of Things Journal*, vol. 7, no. 6, pp. 5159–5170, 2020.

Biography

Ya-Ting Duan received the B.S. degree from the North University of China in 2019. She is currently pursuing the M.S. degree in applied mathematics with the School of Mathematics and Statistics, Shaanxi Normal University, Xi'an, China. Her research interests include security protocols and searchable encryption in cloud storage.

Yan-Ping Li received her M.S. degree from Shaanxi Normal University in 2004 and Ph.D. degree from Xidian University in 2009, Xi'an, China. She now is an Associate Professor with the School of Mathematics and Statistics, Shaanxi Normal University. Her research interests include public key cryptography and its applications.

Lai-Feng Lu received M.S. and Ph.D. degrees in Computer System Architecture from Xidian University, Shaanxi, China, in 2005 and 2012, respectively. Now she is an Associate Professor in Shaanxi Normal University. Her research interests include privacy protection and ad hoc network security.

Kai Zhang received the M.S. degree in Applied Mathematics from Shaanxi Normal University in 2013 and Ph.D. degree in from Xidian University in 2017, Shaanxi, China. Now he is a Lecture in Shaanxi Normal University. His research interests include information security and privacy, and applied cryptography.

A Novel Smart Lock Protocol Based on Group Signature

Yonglei Liu[1], Kun Hao[1], Jie Zhao[1], Li Wang[1], and Weilong Zhang[2]
(Corresponding author: Jie Zhao)
School of Computer and Information Engineering, Tianjin Chengjian University, China[1]
26, Jinjing Road, Xiqing District, Tianjin, China
Quality Management Center, Hebei Jiaotong Vocational and Technical College, China[2]
Email: zhaoj@tju.edu.cn
(Received Mar. 2, 2021; Revised and Accepted July 8, 2021; First Online Dec. 18, 2021)

Abstract

Aiming at security vulnerabilities, complex identity token management, and lack of privacy protections in the smart lock protocol, we propose a novel smart lock protocol based on group signature. As a result, two types of cyber-attacks in existing protocols are discovered: Random number attacks and parallel session attacks. With challenge and response, the improved protocol fixes these two security vulnerabilities by group signature and mutual identity authentication. Furthermore, the complexity of unlocking identity tokens is reduced from O(n) to O(1), and the improved protocol can be applied to anonymous unlocking scenarios. Finally, the indistinguishability game proof and analysis show that the proposed smart lock protocol complete anonymous unlock, satisfy other security requirements of smart lock system such as mutual identity authentication and traceability for identity information of the unlocker, and has certain application prospects in the lightweight IoT smart device market.

Keywords: Anonymity; Group Signature; Mutual Identity Authentication; Provable Security; Smart Lock

1 Introduction

With the development of 5G mobile communication technology, big data, artificial intelligence, and other technologies, as well as the development and popularization of information acquisition and processing technologies such as smart sensors and cloud computing, people can obtain more knowledge from massive data and improve the level of human civilization, human society has also entered the Internet of Things [15] era. As an important derivative concept of the Internet of Things, smart devices [20] refer to equipment, appliances or machines that have sensitive and accurate perception functions, correct thinking and judgment functions, and effective executive functions. Smart devices have brought great convenience to our lives, such as smart home devices [5, 18], Internet of Vehicles [23], wearable devices [9], *etc.*, and have quickly penetrated into various fields of economy and society, involving education, logistics, medical care, automobiles, transportation, construction, *etc.* In 2020, smart device market has reached trillions in China, and smart homes, consumer electronics and smart cars are developing rapidly [6]. As a typical smart device, smart locks are gradually replacing traditional locks and are widely used in smart homes, smart cars, shared bicycles [19] and other fields. The user only needs to approach the smart lock and use the smart phone to unlock it. However, because smart lock manufacturers focus more on function realization, lack of security protection research and development capabilities, or considering development time and cost, there are common security risks in smart lock designs [24].

At present, the deep integration of smart devices with cloud computing and big data to form a data market [11] is the main deployment mode. While improving the level of intelligent control of devices, producers and consumers can trade data to discover user habits and preferences to improve the quality of products and services. But this also brings a new attack surface. Hackers can penetrate cloud servers, traverse and break through the access point of smart devices to connect to the Internet of Things, such as smart gateways [22], infect gateways and smart devices through malicious code, and launch distributed denial of service attacks, *etc.*, which can lead to data privacy leakage, malicious control, and serious consequences such as system failure [10]. Compared with other smart devices, the security of smart locks is more important. Once breached, it will cause huge threats to smart devices, family property and even personal safety, such as burglary after breaking the door lock. Therefore, considering the potential security risks of smart locks online for a long time, the difficulty of smart lock wiring in the door, and the energy consumption and cost of hardware modules, most smart locks in the industry cannot directly connect to the Internet through a fixed gateway, and need to use

Bluetooth Low Energy (BLE) Technology to connect to the Internet through smartphones as a mobile gateway, which is called Device-Gateway-Cloud model (DGC) [3]. The discussion of the security of smart locks in this article is also limited to the DGC model. Although the smart lock system using the DGC model can be offline for a long time to mitigate remote attacks and penetration, there are still serious security threats such as malicious unlocking, man-in-the-middle attacks, state consistency attacks, and even undiscovered security vulnerabilities. Furthermore, some anonymous unlocking scenarios, such as entering a self-serving adult product store are not considered. Therefore, based on previous research works, this article has discovered two security vulnerabilities in the original smart lock protocol, and proposed an improved smart lock protocol based on group signatures, which fixes the security vulnerabilities, simplifies the complexity of the certificate, and support anonymous unlocking.

2 Related Work

This section will summarize and review relevant researches on the security of smart locks.

2.1 Traditional Digital Lock

A digital lock system called Grey project is developed for office environments in 2005 [2]. An equipment-based authentication method is proposed in this project by using early smart phone features, including usage of camera to scan a QR code to obtain public keys and network addresses, mobile phone anti-theft, and proof of access rights. However, the system unlocking delay is relatively high, and smart phones are evolving rapidly. Nowadays, mobile phone features and user needs have changed. Although the gray project is outdated, it has certain guidance for developing smarter locks that are safer, more user-friendly, and faster unlocking. The Raspberry Pi is a tiny and affordable computer, Pinjala *et al.* [13] realize a smart lock based on the Raspberry Pi. After the visitor presses the doorbell, the smart lock activates the camera and sends reminders and real-time images to the administrator's smartphone. The administrator views and remotely authenticates visitor to complete the unlocking.

2.2 Smart Lock with Biometric Authentication

Smart locks with biometric authentication solve the problems of traditional digital lock that passwords are easy to lose and difficult to remember, and currently occupy certain markets, such as fingerprint recognition, face recognition, gesture recognition, and voice recognition. Zhu *et al.* [27] use face recognition technology and open source software OpenCV to propose an attribute tracking algorithm and effectively improve the recognition accuracy. Compared with traditional digital locks, the use of biological features improves the security of remembering and storing keys, but increases the complexity of system deployment, additional hardware, and product costs. At the same time, in view of the application background of the Internet of Things, smart locks with biometric authentication usually lack fine-grained authority management and access control mechanisms.

2.3 Smart Lock with Other Auxiliary Technology

Some smart locks use other communication technologies as auxiliary authentication methods [7], such as audio channel, USB, NFC, VLC, *etc.* Among them, visible light communication (VLC) technology uses visible light as the information carrier, and direct transmission of light signals in the air, which can effectively construct a secure information space with anti-interference and low energy consumption. Song *et al.* [14] use visible light recognition technology to realize a smart lock. After the user uses the smart phone to complete the authentication, the LED light will send out a visible light signal, and the receiver on the smart lock receives the signal to complete the decoding, authentication and unlocking. However, smart phones and LED lights should be connected to the same home gateway, which increases the complexity of system deployment.

2.4 Smart Lock with DGC Model

In order to adapt to the Internet of Things environment and reduce system complexity and control costs, most smart lock manufacturers adopt DGC architecture, such as August, Danalock, Okidokeys, and Kevo. Since the smart lock has no fixed network connection under the DGC model, the user's smart phone is required to act as a mobile gateway. Therefore, if a malicious user forcibly offline the phone, it will cause the communication interruption between smart lock and the cloud server, inconsistency of system status, and revocation evasion. Ho *et al.* [3] use eventual consistency to solve the above problems. As long as an honest user approaches the smart lock, the cloud server will update the user permission list of the smart lock. However, this solution has an attack time window. During the time window, malicious users can still use the recovered credentials to unlock the door before the owner reaches home. In addition, in order to solve the location spoofing in the man-in-the-middle attack, wireless body area network technology is used to indicate the unlocking intention and assist the authentication of the unlocker. Patil *et al.* [12] introduce an additional random number segment in the interaction between the smart lock and the cloud server in order to compensate for the attack time window of the state consistency attack. For users whose unlocking authority has been revoked, the server no longer provides encrypted random numbers for users. Therefore, illegal users will be rejected because

Figure 1: DGC model of smart lock

they cannot provide the random number encrypted by the server within the attack time window. In addition, in order to prevent attackers from maliciously concealing the unlock access logs, additional cameras are used to directly upload the access logs to the cloud server, but the attack surface of the system is undoubtedly increased. Xin *et al.* [21] propose an attribute-based access control mechanism for the problem of cascading deletion of smart lock permissions, using multiple environment attributes to refine access control and support group management, which is applicable to complex family relationships. Bapat *et al.* [1] use steganography to enhance the security of Bluetooth low energy communication between smart phones and smart locks.

In summary, the smart lock with biometric authentication enhances the authentication security to a certain extent, but requires additional hardware and lacks the classification control ability of multiple roles. The auxiliary smart lock also requires additional hardware, which increases the complexity of system deployment. The existing DGC smart locks still have security vulnerabilities. Due to the use of unlock identity token management, as the number of users increases, smart locks have the problem of token storage space overhead, and anonymous unlocking scenarios are not considered, such as entering a self-serving adult product store. Therefore, this article will solve these issues.

3 DGC Model and Security Analysis

This section introduces the DGC-based smart lock system architecture and conducts a security analysis.

3.1 DGC Model

The DGC model consists of three parts: a smart lock installed in the door, a smart phone, and a remote cloud server. Smart locks do not have a direct network connection to cloud services deployed on the Internet, requiring a smart phone to act as a wireless mobile gateway. The smart phone communicates with the smart lock through BLE, and the smart phone communicates with the remote cloud server through mobile communication network such as 5G, as shown in Figure 1.

Users are divided into four categories: owner, resident, recurring guest, and temporary guest. Among them, the owner can unlock the smart lock at any time and use all the administrator functions provided by the smart lock manufacturer, such as granting/recovering permissions, viewing unlock access logs, and updating keys. Residents can unlock the smart lock at any time but cannot use the administrator functions. Recurring guests can unlock the smart lock during the authorized time period, such as housekeepers who come to clean the house every Tuesday from 9 to 11 am. Temporary guest can unlock the smart lock during a temporary period, such as a neighbor visiting from 3 to 4 pm on a sunny day.

3.2 Smart Lock Protocol

The existing smart lock protocol [12] is divided into three phases: Initialization, permission update, and unlocking.

3.2.1 Initialization

- The smart lock (unique identification ID_L) is built with the root key RK_L at the factory, and RK_L is sent to the owner safely and confidentially along with the product manual. At the same time, secure storage (ID_L, RK_L) in the manufacturer's cloud server is processed. And each user has his own public and private key pair (PK_U, SK_U).

- The owner uses RK_L to access into the cloud server and generates the user's unlock identity certificate ($Token_U$), which is stored in the cloud server subsequently. $Token_U = (ID_L, ID_U, SN, Name_U, Type_u, Days_U, Time_U, Dates_U, PK_U)_{RK_L}$, Wherein, $Name_U$ is the user's name; ID_U is the unique user identification, which can be the user's mobile phone number; SN is the sequence number of the unlock identity certificate for permission update; $Type_U$ is the user type such as owner, recurring guest, *etc.*; A combination of $Days_U$ and $Time_U$ describes the authorized unlocking times for recurring guests, $Dates_U$ describes the authorized unlocking times for temporary guests. $Token_U$ is encrypted by RK_L.

- The owner sends $Token_U$ to the user, or after the user accesses the cloud server and passes identity authentication, such as a short message SMS, the user downloads the $Token_U$ in the smart phone.

3.2.2 Permission Update

- The owner access into the cloud server and updates the user's token.

- The owner's smartphone enters the Bluetooth communication range of the smart lock, and the cloud server sends the updated $Token'_U$ to the smart lock.

- The smart lock uses RK_L to decrypt $Token'_U$, verifies whether the SN is fresh, and updates the permission list.

3.2.3 Unlocking

- When the user's smart phone enters the Bluetooth communication range of smart lock, the user sends the $Token_U$ stored in the smart phone to the smart lock.
- The smart lock uses RK_L to decrypt $Token_U$, verifies whether the authority is consistent with the local tokens stored in the smart lock, and obtains the user's public key RK_U, and then (ID_L, N_1) is encrypted by PK_U and is sent to the user.
- The user uses the private key SK_U to decrypt to received (ID_L, N_1), verifies the ID_L and sends N_1 to the smart lock.
- The smart lock verifies N_1 and unlocks. The unlock access log is encrypted by RK_L and is sent to the cloud server.

3.3 Attack Behavior Analysis

Since smart locks with DGC model belong to distributed system, which is inevitably with a network partition. According to the CAP theorem, the availability and consistency of access permission lists, unlock access logs and other data in smart locks and cloud servers cannot be realized at the same time. Therefore, the most serious threat of smart locks is state consistency attacks, such as revocation evasion, access log evasion, *etc.* In addition, due to the characteristics of BLE communication, there are man-in-the-middle attacks and denial of service attacks against the BLE protocol itself. In this section of attack behavior analysis, we analyze the previous research work about state consistency and man-in-the-middle attacks in Section 3.3.1 and 3.3.2. The Sections 3.3.3 and 3.3.4 are our novel work. We discovered random number attacks and parallel session attacks against security vulnerabilities in the original smart lock protocol.

3.3.1 State Consistency Attack

- Revocation evasion. The owner access into the cloud server to withdraw the attacker's unlocking authority, and then the owner with the smart phone approaches to the smart lock to update the permission. However, in the attack time window, the attacker can still unlock the smart lock before the owner reaches home.
- Access log evasion. After the attacker unlocked the smart lock, the malicious smartphone receives the unlock access log and refuses to forward to the cloud server. In this way, the attacker could claim that stealing after unlocking is not convicted because the log was lost.
- Threat mitigation. Ho *et al.* [3] use eventual consistency to solve the above problems, all honest users regardless of owner unlocking the lock, it will trigger the permission update operation, but this action still has an attack time window. In addition, the unlock access log sent by the smart lock should be confirmed by the cloud server's signature, otherwise it will be retransmitted. Therefore, even if the attacker blocks the forwarding, the access log will be eventually uploaded to the cloud server when the honest user unlocks the smart lock. However, the attack time window still exists. Patil *et al.* [12] introduce a random number N_c between the smart lock and the cloud server in order to defend revocation evasion. In the final step of unlocking phase, the user sends N_c encrypted by the cloud server using RK_L. The smart lock cannot be unlocked within the attack time window as the encrypted N_c is not be provided by the cloud server. However, the unlocking process requires the participation of the cloud server, in case of network connection problems or cloud server failure, the smart lock cannot be unlocked and the system availability cannot be guaranteed. In addition, in order to defend access log evasion, an additional camera with permanent network connection is used to directly upload the access log to the cloud server, but this undoubtedly increases the attack surface of the system.

3.3.2 Man-in-the-Middle Attack

- The man-in-the-middle attack of the smart lock is essentially a collusion attack. Attacker A and attacker B have BLE communication channels, and both parties establish a hidden tunnel connection. A is close to the smart lock and is paired through BLE, and B is close to the user's smartphone and is paired through BLE. The user will mistakenly find that there is a smart lock around and sends token to B. B receives the token and forwards to A through the hidden tunnel. A sends the token to the smart lock. The smart lock generates a challenge. A continues to forward the challenge to B through the hidden tunnel, and B sends the challenge to the user. The user generates the challenge response and sends to B, and then B forwards the challenge response to A through the tunnel. Finally, A sends the challenge response to the smart lock, and the unlock succeeds.
- Threat mitigation. Some commercial smart locks use geo-fencing [16] technology to assist the user's mobile phone to determine whether the smart lock is nearby. Therefore, the user's smartphone can recognize that the attacker B is not a real smart lock, and refuses to send token to initiate the session. However, studies have pointed out that geo-fencing technology has security threats such as mistaken unlocking in multiple entrances and exits scenario, and there are also geo-fencing spoofing attacks. The root of the problem lies in auto-unlocking. If the user's unlock intention is asked every time, such as APP pop-up

window, SMS, *etc.*, man-in-the-middle attacks can be easily identified. However, it greatly reduces the user experience and brings boredom. Therefore, the existing improvements are mainly to balance user experience and safety. Kevo smart lock combines geo-fencing and touch unlocking technology, the user re-enters the geo-fencing boundary and touches the unlock button to unlock. However, the system cannot verify the identity of the toucher. Ho *et al.* [3] use wireless body area network technology to indicate the unlocking intention. Compared with the Kevo smart lock, after the user touches the unlock button, the smart lock and the user's wearable device will complete the authentication to confirm the user's identity and then unlock.

3.3.3 Random Number Attack

- We find an attack called random number attack. In the last step of the unlocking phase, the user sends N_1 to the smart lock in plaintext. Therefore, the attacker passively listens for all communications between the smart lock and the legitimate user. And then, the attacker initiates a new session and replays the messages acquired during the passive listening. If the N_1 is predictable, the attack can be succeeded, for instance, the N_1 is incremental, the attacker injects $N_1 + 1$ to maliciously unlock.

- Threat mitigation. The random number N_1 is not predictable or is transmitted encrypted. In addition, the impersonation attack is fixed according to Section 3.3.4.

3.3.4 Smart Lock Impersonation Attack

We find a smart lock impersonation attack launched by parallel session attack. The existing smart lock protocols mainly focus on authentication of user rather than the authentication of smart lock, because even if user unlock a fake smart lock, none of assets are lost. For instance, the original smart lock protocol only uses ID_L to authenticate the smart lock, the attacker can easily predict ID_L through manufacturer past product identification number and product information.

- Attack description. Although the PK_U is protected in $Token_U$ and encrypted by RK_L, compared to private key, the PK_U is more easily to disclosed. Therefore, the attacker can predict ID_L and acquire PK_U in order to impersonate a legitimate smart lock. And then, the attacker launches unlocking interaction with legitimate users and obtains all communication traffic for replay attack, offline RK_L cracking and user unlocking behavior analysis. Under the conditions that the random number N_1 is not predictable and is transmitted encrypted, we find a new parallel session attack to make random number attack successful. Firstly, we launch random number attack and acquire (ID_L, N_1) encrypted by PK_U. Secondly, without SK_U, we cannot acquire N_1, then launch a parallel session attack to impersonate a legitimate smart lock to ask the oracle of N_1 from a legitimate user. Finally, the response of N_1 is sent to victim of smart lock to complete malicious unlocking. The attack details are showed in Figure 2.

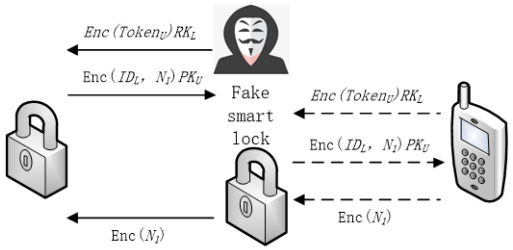


Figure 2: Parallel session attack

- Threat mitigation. The smart lock and the user are authenticated by a secure mutual authentication protocol, such as a challenge-response based mutual authentication protocol.

4 Improved Smart Lock Protocol based on Group Signature

The existing smart lock has high storage and communication overheads of unlocking identity certificate. In addition, in some scenarios, users have a strong desire to protect privacy, such as entering a self-serving adult product store. And as mentioned in Section 3.3, we found random number attacks and parallel session attacks. In response to the above problems, we propose a novel smart lock protocol to improve the smart lock protocol of reference [12] and use group signature with a shorter signature length [4] in our improvement protocol to realize the above security goals, as well as, suitable for lightweight smart home devices [8] such as smart locks. Detailed group signature algorithm can be found in the literature [4]. The improved protocol in this article focuses on anonymity, reducing communication and storage overheads of lengthy unlocking identity credential, and the security feature of the protocol to resist random number attacks and parallel session attacks that we have discovered.

4.1 Group Signature

4.1.1 Setup

The algorithm is given a system safety parameter λ and generates system public parameters $PP = (q, G_1, G_2, G_T, e, P_1, P_2, H(\cdot), n)$, wherein, G_1, G_2, and G_T are cyclic groups of order q (of length λ bits). $e : G_1 \times G_2 \rightarrow G_T$ is a bilinear mapping. P_1 and P_2 are generators of G_1 and G_2 respectively. $H(\cdot) : \{0,1\}^* \rightarrow \{0,1\}^n$is a safe hash function. The random numbers d, s, u are

chosen to calculate $D = d \cdot P_1, S = s \cdot P_2, U = u \cdot P_1$ and respectively. The private key of group administrator sk is (d, s); the tracing private key tk is u; the group public key gpk is (D, S, U).

4.1.2 Enroll

According to the PP and the group administrator sk, the group member private key gsk is generated. x_i is randomly select and calculate $Z_i = (d - x_i)(sx_i)^{-1} \cdot P_1$ and $tag_i = H(x_i \cdot Z_i)$. The private key of the group member $gsk_i = (x_i \cdot Z_i)$. The group administrator manages the list of members $list = (GU_i, tag_i)$.

4.1.3 GSign

According to PP, gsk_i, gpk, group members randomly select k and calculate $C_1 = k \cdot P_1, C_2 = x_i \cdot Z_i + k \cdot U$ and $Q = e(U, S)^k$. For the message m to be signed, the group members further calculate $c = H(C_1, C_2, Q, m)$ and $w = kc + x_i$. The signature $GSig(m) = (C_1, C_2, c, w)$.

4.1.4 GVerify

The verifier verifies whether the signature is legal according to m, $GSig(m)$, and gpk. The verifier calculates $Q' = \frac{e(C_2,S) \cdot e(P_1,P_2)^w}{e(c \cdot C_1 + D, P_2)}$ and verifies whether $c = H(C_1, C_2, Q', m)$.

4.1.5 GTrace

The group manager determines the signer according to the tracing private key tk and $GSig(m).tag_i = H(C_2 - u \cdot C_1)$ is calculated and search to determine the signer in the list of members $List$.

4.1.6 Revoking

The group administrator adds the revoked member's private key material x_i or $x_i \cdot Z_i$to the revocation list $RList$, the verifier traverses all the x_i or $x_i \cdot Z_i$ in the $RList$, and recognizes the revoked member by whether $e(C_2, S) \cdot e(x_i \cdot P_1, P_2) = e(D, P_2) \cdot Q'$ or $e(C_2 - x_i \cdot Z_i, S) = Q'$ respectively.

4.2 Improved Smart Lock Protocol

Due to space limitations, this section focuses on describing the differences and improvements compared to the original smart lock protocol. Wherein, the details of unlocking are shown in Figure 3.

4.2.1 Initialization

- The smart lock generates its own public-private key pair (PK_L, SK_L). The users are managed by group including permanent user group and guest group, and the token is simplified. The owner/administrator creates a group signature system to generate one permanent user group and n guest groups with GPK, tk, and sk. The owner/administrator uses sk to generate gsk of each user. $Token = (ID_L, SN, GN, Time, Cycle, gpk)_{RK_L}$. Wherein, GN is the group number, 1 represents the permanent user group, and $2-n$ represents the guest group with overlapping visit times, such as cleaning staff and water and electricity maintenance workers who visit from 8:00 to 9:00 every Tuesday. gpk is group signature public key. Given that guests often visit periodically, such as weekly cleaning and annual house maintenance, the field of the $Cycle$ is made to replace the field of $Days$ and $Dates$ for simplifying and reducing the length of token in original smart lock protocol. For instance, $Cycle$ is 0 for temporary visit once, 1 for every day, 30 for every month, *etc.* And access time determination is whether current time is equal to $Time + nCycle$.

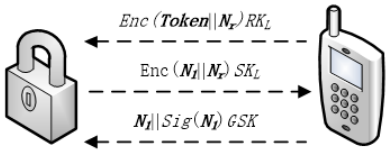


Figure 3: Improved unlocking protocol

- The owner/administrator securely sends the $Token$ and the corresponding group member private key gsk to each user.

  Compared with the original protocol, all users from one group only store one token on the smart lock and cloud server, where n is the number of users, the complexity is reduced from O(n) to O(1). And the length of the token is also reduced, which is more friendly to lightweight IoT devices such as smart locks.

4.2.2 Unlocking

- When the smart phone of user enters the scope of Bluetooth communication with smart lock, the user sends $Token||N_r$ to the smart lock.
- The smart lock uses RK_L to decrypt $Token$, and determines freshness and updates $Token$ through SN; the smart lock verifies whether the permissions are consistent with the local Token stored in the smart lock; the $Token$ will be stored in the smart lock directly, if a user of the group opens the lock the first time; the smart lock acquires the group signature public key GPK. (N_1, N_r) is encrypted by the private key SK_L of smart lock and sent to the user.
- User decrypts (N_1, N_r) by the public key PK_L of the smart lock, verifies the N_r, sends N_1 to the smart lock and signs N_1 with the private key GSK of the group member.
- The smart lock verifies N_1, uses GPK to verify the signature, and judges whether the user's permission

has been revoked according to the group member revoking algorithm. If the above verifications pass, the lock will be unlocked, and the unlock access log is sent to the cloud server.

4.2.3 Permission Update and Log Access

- The owner/administrator accesses into the cloud server and generates the RList using the group member revoking algorithm.

- The smart phone of the honest user enters the Bluetooth communication range of the smart lock, and the cloud server sends encrypted RList to the smart lock.$(ID_L, SN, GN, RList)_{RK_L}$

- If the above algorithms are performed or partly performed on the smart door lock or the user's mobile phone, that the smart door lock is directly controlled by the mobile phone, more functions and class libraries related to smart lock hardware should be installed on the smart phone, which increases the complexity of the system and the energy consumption of the smart lock undoubtedly. Moreover, for the equipment manufacturer, the control of the smart lock and the collection of product usage data are lost. Under the DGC model, the wireless communication link such as 5G from the mobile phone to the cloud server has more mature security protection methods, but the BLE link between the mobile phone and the smart door lock is relatively insecure, and there are more security threats such as illegal eavesdropping and man-in-the-middle attacks. Therefore, in the balance between cloud server single point of failure and system security, this paper also continues to use the DGC model for group signature algorithm performed in could server.

- Two methods can be adopted to trace unlocker. The former one is that the smart lock sends unlock access log with group signature to the cloud server, the owner/administrator logins cloud server and use tk to trace unloker. The latter one is that the cloud server sends $(ID_L, SN, GN, RList)_{RK_L}$ to the smart lock in the initialization phase, the smart locks use tk trace unlocker and sends unlock access log with revealed unlocker directly to the cloud server. This approach conflicts with the original intention of anonymity and increases the security risk of the system when the smart lock is physically attacked or captured, so we adopt the former.

4.3 Security Analysis

In this section, using the provable security theory, a security model of our improved smart lock protocol is established, and security analysis is processed.

4.3.1 Adversary Model

The system has smart locks, users, cloud servers, and adversaries A. The ability of the adversary A is consistent with the Dolev-Yao model [17], which can passively listen, steal, forge, and block all communication between users in the channel. Using an oracle to simulate an instance run of the protocol, the attack capabilities of adversary A can be simulated as the following oracle queries:

- Setup-Oracle. A can acquire PP, gpk and public key PK_L of the smart lock by querying the oracle.

- Excute-Oracle. The oracle simulates passive attacks. A can acquire all the messages sent by honest users running the smart lock protocol by querying the oracle.

- Send-Oracle. The oracle simulates active attacks, A sends a message msg to the oracle, and the oracle processes msg according to the protocol rules, and sends the result msg' to A.

- Corrupt-Oracle. The oracle simulates the loss of the unlocking identity credential, and A can acquire the token of any user through query.

- Handle Dispute-Oracle. The oracle simulates signature tracing, and A asks the identity of the signer associated with the N_1. The oracle calculates tag_i through tk and returns tag_i to A.

- Test-Oracle. The oracle does not simulate attack ability of A, but judges the advantage of A in winning the game. After receiving the query request, A randomly selects two unlocking sessions S_0 and S_1 that A has never inquired in the handle dispute oracle. The oracle randomly selects $b \in \{0, 1\}$, and uses tk to trace the message in the session Sb: $N_1||GSig(N_1)$ and calculate tag_b. A gets tag_b and guesses b'. If $b' = b$, A wins, otherwise, A loses. $ADV^A_{SL}(A)$ is defined as the winning advantage of A.

$$ADV^A_{SL}(A) = |Pr(b' = b) - 1/2| \tag{1}$$

4.3.2 Indistinguishability Game Proof

Theorem 1. *The improved smart lock protocol is anonymous, if and only if $ADV^A_{SL}(A)$ is negligible for any polynomial adversary A.*

Proof. Assumes that $ADV^A_{SL}(A)$ is not negligible in distinguishing unlock session S_0 and S_1. Defines adversary B that can break the group signature system, and the attack capabilities of adversary B can be simulated by similar oracles in Section 4.3.1, wherein, in the Test-Oracle, B randomly selected two group signature message $m_1||GSig(m_1)$ and $m_2||GSig(m_2)$. The oracle randomly selects $b \in \{0, 1\}$, and uses tk to trace the message $m_b||GSig(m_b)$ and calculate tag_b. B gets tag_b and guesses b'. If $b' = b$, B wins, otherwise, B

loses. Obviously,$ADV_{SL}^{A}(A) = ADV_{GS}^{A}(B)$ then the adversary B has a non-negligible advantage in breaking the anonymity of group signature, which contradicts with the Theorem that there is no polynomial adversary can attack the group signature anonymity with a non-negligible advantage [4], so the hypothesis is not valid and Theorem 1 is correct. □

4.3.3 Other Security Analysis

In addition to unlocking anonymity, the improved smart lock protocol has other security features.

- Resistance to state consistency and man-in-the-middle attacks. (in Section 3.3.1 and 3.3.2) Aiming at state consistency attacks and man-in-the-middle attacks, we follow the improvements of previous researchers. Due to system complexity considerations, we use the honest user mechanism and log confirmation mechanism of Ho *et al.* [3] to mitigate state consistency attacks, and use the mechanism of disabling auto-unlocking by the advisement of Patil *et al.* [12] to mitigate man-in-the-middle attack in our improved protocol.

- Mutual authentication for resistance to random number and parallel session attack. (in Section 3.3.3 and 3.3.4)The improved smart lock protocol uses challenge N_r and the smart lock private key SK_L to encrypt N_r. Since the attacker cannot obtain SK_L, the smart lock cannot be faked. Simultaneously, a group signature is used for user authentication. Since the attacker cannot obtain the group member's private key gsk_i, the signature of N_1 cannot be constructed for the attacker to impersonate the user. Therefore, the improved smart lock protocol can resist random number attacks and parallel session attacks.

- Token loss and forgery. If the user's Token is illegally leaked, although the attacker can initiate an unlock request, but since the group member's private key gsk_i cannot be obtained, the signature of N_1 cannot be constructed to impersonate the user. In addition, because the attacker cannot obtain the root key RK_L and cannot forge $Token$.

- Resistance to replay attacks. When an attacker performs a replay attack, it will fail because of the freshness of the random number, even if using the Send-Oracle in 4.3.1. For example, the attacker C impersonates a legitimate user to maliciously unlock the lock and replay the unlock request. In last step of the protocol, C requires a group signature on the random number selected by the smart lock, such as N_6. C launches a parallel session attack through Send-Oracle and impersonates a legitimate smart lock to send N_6 to a legitimate user for the answer to the N_6 signature. But since the private key SK_L of the smart lock cannot be obtained, the Send-Oracle cannot be performed. Even if the attacker replays the question N_6 encrypted SK_L, however the N_r of two parallel session are different, the attack is impossible to succeed. Therefore, C can only passively listen to the communication traffic of the legitimate user and smart lock for a long time until C captures the N_6 signature and replays.

  Assumes that $|N_1|$ represents the length of random number N_1. Obviously, the following winning advantage of adversary C can be conducted.

$$ADV_{SL}^{Replay}(C) \leq 1/2^{|N_1|} \tag{2}$$

  Under a secure random number length, such as 256 bits, the winning advantage of adversary C can be ignored, and the improved smart lock protocol can resist replay attacks.

- Traceability. When a security accident or property loss occurs, the owner/administrator can use the tracing key tk to find the intruder. Since the attacker cannot obtain tk, the attacker cannot acquire traceability.

5 Conclusions

With the advent of the Internet of Things era, smart devices have gradually entered all aspects of social and economic life. As an important smart device, the security of smart locks has attracted much attention. Two new security vulnerabilities in the existing DGC architecture smart lock protocol are discovered in this article: random number attacks and parallel session attacks. A smart lock protocol based on group signature is proposed, which simplifies the complexity of unlocking identity credentials from O(n) to O(1), resists random number attacks and parallel session attacks, and can be applied to anonymous unlocking scenarios. The indistinguishability game proof and security analysis show that the improved smart lock protocol proposed in this article satisfies the security requirements of smart locks and has certain application prospects. In the next step, we will study the proposed improved smart lock prototype system and explore the combination of other related security technologies, such as machine learning [26] and blockchain [25].

Funding: This research was funded by National Natural Science Foundation of China, grant number NO. 61902273.

References

[1] C. Bapat, S. Inamdar, G. Baleri, *et al.*, "Smart-lock security re-engineered using cryptography and steganography," in *International Symposium on Security in Computing & Communication*, pp. 325-336, 2017.

[2] L. Bauer, S. Garriss, J. M. Mccune, *et al.*, "Device-enabled authorization in the grey system," in *International Conference on Information Security*, pp. 431-445, 2005.

[3] G. Ho, D. Leung, P. Mishra, *et al.*, "Smart locks: Lessons for securing commodity internet of things devices," in *Proceedings of the 11th ACM on Asia Conference on Computer and Communications Security*, pp. 461-472, 2016.

[4] T. Ho, L. Yen, C. Tseng, "Simple-yet-efficient construction and revocation of group signatures," *International Journal of Foundation of Computer Science*, vol. 26, no. 5, pp. 611-624, 2015.

[5] Q. Huang, Z. Li, W. Xie, *et al.*, "Edge computing in smart homes," *Journal of Computer Research and Development*, vol. 57, no. 9, pp. 1800-1809, 2020.

[6] iiMedia Research, *2020 China Smart Hardware Industry Development Panorama Research Report*, 2020. (`https://www.iimedia.cn/c400/70397.html`)

[7] J. Jeong, "A study on smart door lock control system," *Cluster Computing*, vol. 19, no. 3, pp. 1-11, 2016.

[8] C. Lin, D. He, N. Kumar, *et al.*, HomeChain: A blockchain-based secure mutual authentication system for smart homes," *IEEE Internet of Things Journal*, vol. 7, no. 2, pp. 818-829, 2020.

[9] Q. Liu, T. Li, Y. Yu, *et al.*, "Data security and privacy preserving techniques for wearable devices: A survey," *Journal of Computer Research and Development*, vol. 55, no. 1, pp. 14-29, 2018.

[10] Y. Meng, S. Li, Y. Zhang, *et al.*, "Cyber physical system security of smart home platform," *Journal of Computer Research and Development*, vol. 56, no. 11, pp. 2349-2364, 2019.

[11] M. Moniruzzaman, S. Khezr, A. Yassine, *et al.*, "Blockchain for smart homes: Review of current trends and research challenges," in *Computers & Electrical Engineering*, vol. 83, 2020. (`https://doi.org/10.1016/j.compeleceng.2020.106585`)

[12] B. Patil, P. Vyas, R. K. Shyamasundar, "SecSmartLock: An architecture and protocol for designing secure smart locks," *International Conference on Information Systems Security*, pp. 24-43, 2018.

[13] S. Pinjala, S. Gupta, "Remotely accessible smart lock security system with essential features," in *International Conference on Wireless Communications Signal Processing and Networking*, 2019. DOI: 10.1109/WiSPNET45539.2019.9032715.

[14] S. J. Song, H. Nam, "Visible light identification system for smart door lock application with small area outdoor interface," *Current Optics and Photonics*, vol. 1, no. 2, pp. 90-94, 2017.

[15] Q. Sun, J. Liu, S. Li, *et al.*, "Internet of things: Summarize on concepts, architecture and key technology problem," *Journal of Beijing University of Posts and Telecom*, vol. 33, no. 3, pp. 1-9, 2010.

[16] S. Tang, Y. Yu, R. Zimmermann, *et al.*, "Efficient geo-fencing via hybrid hashing: A combination of bucket selection and in-bucket binary search," *ACM Transactions on Spatial Algorithms & Systems*, vol. 1, no. 2, 2015.

[17] Z. Tang, X. Li, "The formalization description of the dolev-yao intruder model," *Computer Engineering & Science*, vol. 32, no. 8, pp. 36-45, 2010.

[18] J. Wang, Y. Li, Y. Jia, *et al.*, "Survey of smart home security," *Journal of Computer Research and Development*, vol. 55, no. 10, pp. 2111-2124, 2018.

[19] S. Wang, Y. Liu, D. Li, *et al.*, "A ferrofluid-based planar vibration energy harvester for smart lock of shared bicycle," *International Journal of Applied Electromagnetics and Mechanics*, vol. 61, no. 2, pp. 293-300, 2019.

[20] Y. Wang, C. Zhang, D. Huo, *et al.*, "A survey of security threats and defending technologies on IoT smart devices," *Journal of Cyber Security*, vol. 3, no. 1, pp. 48-67, 2018.

[21] Z. Xin, L. Liu, G. Hancke, "AACS: Attribute-based access control mechanism for smart locks," *Symmetry*, vol. 12, no. 6, pp. 1050, 2020.

[22] W. Yan, Z. Wang, H. Wang, *et al.*, "Survey on recent smart gateways for smart home: Systems, technologies, and challenges," in *Transactions on Emerging Telecommunications Technologies*, 2020. (`https://doi.org/10.1002/ett.4067`)

[23] F. Yang, S. Wang, J. Li, *et al.*, "An overview of internet of vehicles," *China Communications*, vol. 11, no. 10, pp. 1-15, 2014.

[24] M. Ye, N. Jiang, H. Yang, *et al.*, "Security analysis of internet-Of-things: A case study of August smart lock," in *IEEE Conference on Computer Communications Workshops*, 2017. DOI: 10.1109/INFCOMW.2017.8116427.

[25] Y. Yuan, F. Wang, "Blockchain: The state of the art and future trends," *Acta Automatica Sinica*, vol. 42, no. 4, pp. 481-494, 2016.

[26] L. Zhang, Y. Cui, J. Liu, *et al.*, "Application of machine learning in cyberspace security research," *Chinese Journal of Computer*, vol. 41, no. 9, pp. 1943-1975, 2018.

[27] Z. Zhu, Y. Cheng, "Application of attitude tracking algorithm for face recognition based on openCV in the intelligent door lock," *Computer Communications*, vol. 154, pp. 390-397, 2020.

Biography

Yonglei Liu is an associate professor in the school of computer and information engineering, Tianjin Chengjian University. In 2014, he received the Ph.D. degree from Tianjin University in computer application technology. His main research interests include computer network performance optimization, wireless network security, network

protocol security analysis and network security evaluation.

Kun Hao received her M.S. degree from Tianjin University, Tianjin, China in 2006. In 2010, she received her Ph.D. degree in the School of Computer Science and Technology, Tianjin University. She is an Assoc. Prof. in the School of Computer and Information Engineering at Tianjin Chengjian University. Her research interests lie in underwater sensors network, wireless communications and networking, wireless sensor networks, network protocol and network optimization, and application of VR technology in architecture design.

Jie Zhao is an associate professor in the department of electronic information engineering, Tianjin Chengjian University. In 2015 he received the Ph.D. degree from Tianjin University in information and communication engineering, China. Since 2009, he has been working in the school of computer and information engineering, Tianjin Chengjian University. His current research interests include multimedia information security and computer vision.

Li Wang received PhD degree in optics from Nankai University. She currently is a lecturer of school of computer and information engineering, Tianjin chengjian university. Her research interests include the Internet of Things engineering and optical fiber sensing.

Weilong Zhang is an associate professor in the department of electrical and information engineering, Hebei Jiaotong vocational and technical college, master's degree. His current research interests include multimedia information security and near end wireless network communication technology.

Video Steganography for Image and Text Using Deep Genetic Algorithm and LSB

Nouran Mohamed Selim[1], Shawkat Kamal Guirguis[2], and Yasser Fouad Hassan[1,3]
(Corresponding author: Nouran Mohamed Selim)
Computer Science Department, Faculty of Science, Alexandria University[1]
Alexandria, Egypt
Email: nouran.selim@alexu.edu.eg
Information Technology Department, Institute of Graduate Studies and Research, Alexandria University[2]
Faculty of computer science and AI, Pharos University[3]
(Received Jan. 30, 2021; Revised and Accepted July 5, 2021; First Online Dec. 18, 2021)

Abstract

Data security is a threat for delivering data through the internet, and it is concerned with protecting sensitive data from unauthorized access. The security threats increase with the increase of sensitive data, so security issues should be considered because hackers may use vulnerabilities over public networks to thieve the information. Video steganography is considered an efficient technique and becomes an important research area for data security. This paper aims to embed images and text into a video. The proposed algorithm selects the best frames and pixels for embedding images and the best video tags to hide text. As a result, the video's size and visual quality are not altered even after embedding secret data.

Keywords: Encryption; Flash Video; Genetic Algorithm; LSB Technique; Steganography

1 Introduction

Steganography comes from the Greek words which divide into two parts, the first part is "steganos" means covered or concealed and the second part is "graptos" means writing. Steganography is defined as hiding the existence of messages in a particular medium "cover-medium" such as audio, video, image, text communication [13]. Steganography is the art and science of secret communication which conceals the very existence of communication [16]. It is defined as the process of embedding a secret data (message, image or audio) to be hidden in a cover-medium to reproduce stego-medium that no one apart from the sender and receiver even realizes that there is a hidden message.

The basic steganography algorithm of embedding is shown in Figure 1, Steganography hides the data and the fact of communication. It ensures the anonymity of the communication parties. The amount of information transmitted is greater than the secret encrypted information. It needs an additional carrier. A good steganography algorithm should be measured by embedded data as much as possible (embedding capacity), and the perceptual distortion of the cover medium after the embedding process should be minimum as possible (invisibility) [3].Steganography is the procedure of covertly without changing its quality it inserts information inside a data source. In the major section, while concealing the information, Original data is not retained in its unique format. Steganography depends on hiding an undercover message in unsuspected different media information and is by and largely used as a part of secret correspondence between recognized gatherings [1]. In contrast to cryptography, which is the art of protecting secret data by transforming it into an unreadable format, it does not hide the data or hide the fact of communication so no need to any additional carrier. Cryptography has become a basic requirement of public electronic connectivity to secure data during transmission against the possibility of message eavesdropping and electronic fraud [7].

This paper will focus on developing one system that uses both Cryptography and Steganography for more confidentiality and security [18]. Advanced Encryption Standard (AES) algorithm is a very secure technique for cryptography, which is characterized by its flexibility and simplicity. The amount of information transmitted in the communication process depends on the amount of the encrypted information. Both transform information into a form that is incomprehensible to a third party, use the secret key to encrypt or decrypt data that becomes more secure. It is used when communicating over an untrusted medium such as the internet, where information needs to be protected from other third parties. We can apply steganography on text, image, audio or video. In this paper, we hide secret data inside the video file. We select video as cover-medium for reasons like larger spaces of the video in hiding or embedding data and can be embedded

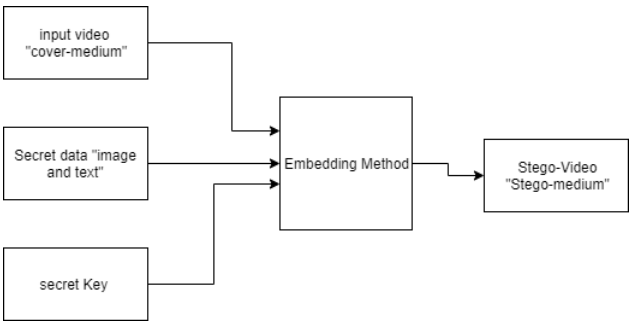


Figure 1: The basic steganography embedded algorithm

data in video/audio tags. Videos are considered as collections of images and sound files which make some of the effective methods of steganography on images and audio files also possible for applying in video files.

Larger space for embedding and having small unnoticeable distortions make video steganography a reliable method in hiding data [12]. Also, we select Flash video format (FLV) because it has smaller file size compared to all the other formats. FLV is very simple. It starts with the headers then metadata tag (data that describes the FLV), then interleaved audio and video tags (actual data). This paper will use two methods to embed image and text in cover video. One of these methods is the least significant bit (LSB) in the pixel/video tags value of the video. It works by replacing the least significant bit of some randomly selected pixels or tags in the cover image with value of new secret data. Another method is embedded secret data with Genetic Algorithm (GA) by calculating the highest fitness function which depends on the lowest difference between pixel of cover image and secret data. Artificial Intelligence (AI) including machine learning, Genetic Algorithm, heuristic optimization is one sub branch of computer science, which can be used to steganography technology and can achieve high visual quality, robustness, low cost, optimal and adaptive solutions. Recently, AI technology is rarely used in video steganography, though applied to various kinds of image steganography, including Genetic Algorithms. Due to the generality of image steganography and pre-embedding video steganography, the AI technology applied to image steganography has great reference value for pre-embedding video steganography [20].

The Rest of the paper is organized as follows: Section 2 discusses related work of steganography. Section 3 presents the proposed work. Section 4 gives results of the work and Section 5 concludes the papers.

2 Related Works

Sahu and Mitra [17] described an algorithm to hide the message in the frames of the video. The algorithm suggested selecting a random video frames then using a pixel swapping algorithm for blue channel of the frames. The secret message is encrypted using AES technique then embedded it into video frames with Least Significant Bit (LSB) technique. Authors conclude that using only LSB technique for data hiding is not a secure method therefore, they was used the random frames selection algorithm and pixel swapping algorithm to improve the security of the method.

Eltahir *et al.* [5] described an algorithm to hide secret data in video by splitting the digital video file into separate frames. They suggested algorithm [9] is based on LSB technique but using a 3-3-2 approach. The 3-3-2 approach uses least significant bits of RGB (Red Green- Blue) level but it takes 3 bits of the red and green and only 2 bits from Blue color to form one byte. They only take 2 bits from blue color because they depend on HVS (Human Vision System) that is more sensitive to blue color. The algorithm produces image look visually like the original one.

Ibrahim *et al.* [8] described an algorithm to conceal data in video frames. They selected video frames and splitted them into three bands (red, green and blue), then applied discrete cosine transform (DCT) and ZigZag scan to convert image from two-dimensional array form to one-dimensional array and after that they sort it from low to high frequency who converted the secret image to binary form then the secret data is embedded using LSB in high frequencies to get little distortion places.

Sudeepa *et al.* [19] proposed an algorithm to hide secret information in cover video based on randomization, Steganography, Symmetric encryption and parallelization. They designed an algorithm which selects random frames and split secret data into parts then apply encryption and embedding technique for each part in parallel which it takes less computational time.

Manisha and Sharmila [11] described an algorithm for hiding secret image within one frame of AVI video. They did a two level of encryption process that uses only two bit positions in a particular video frame to accommodate bits of a secret image and it is placing in four different quadrants. The Size of secret image must be compatible with size of the video frame. They proposed to use a hashing function to hash the bits of the secret image onto the video frame.

Limkar *et al.* [10] proposed an algorithm to hide secret information behind audio and video files. Dividing secret data and video/audio file into frames then embedding secret data in frames and dividing resulting frames into bits and encrypted them using Rivest -Shamir-Adleman RSA/Data Encryption Standard DES/Triple Data Encryption Standard 3DES algorithm. The algorithms try to increase the level of security by encrypted video/audio file after embedded secret information.

Dasgupta *et al.* [4] proposed architecture for hiding information in video frames using 3-3-2 LSB for embedding technique and a genetic algorithm is used as an optimization technique. The optimizer is trying to optimize the stego-frame using the objective function then take the optimized value and goes through the Anti-steganalysis test module. Genetic algorithm is trying to optimize hiding

process in video.

Khodaeia *et al.* [9] proposed a new adaptive steganographic technique to hide secret data within a gray-scale cover image by dividing the cover image into several non-overlapped blocks with two consecutive pixels and then producing the number of secret bits that could embed into two consecutive pixels. They embedded the secret bits into the cover image by modifying the LSBs of two consecutive pixels.

3 Proposed Steganography Method

In this section, the proposed steganography algorithm will try to embed image and text inside the cover video without a third party suspected in.

3.1 Hiding Data Algorithm in Video

This algorithm will discuss how to hide image and text message in video using deep genetic algorithm and Least Significant Bit LSB method. Figure 2 shows the proposed embedding algorithm for hiding image and text in the video. This video called a "cover video" because it is selected to cover or hide all secret data "image and text". The cover video was splitted into frames. In the proposed method, we are using Genetic Algorithm because it is one of the best search techniques that are used to find an optimal or near-optimal solution to the complicated problems [4]. The sender will apply a genetic algorithm to the frames to get the highest fitness function which is calculated by the highest difference between video frames. The reason behind selecting the highest difference between frames is to be less noise sensitive when embedding the secret data and this is described in Equation (1).

$$fitness[x] = max[frame[i] - frame[i+1]]. \quad (1)$$

Applying noise mutation to the frames (that sender is selected from previous step) to get the best frame that is calculated by the lowest affected frame after noise mutation. The sender takes the selected frames with high fitness function $(hF1, hF2, \cdots, hFn)$ to embed parts of the secret image in them. Taking into consideration, the numbers of frames selected are equal to the number of image parts, to put each part of the image in different frame. Also, we must consider the minimum number of the secret image to be divided is four parts and the maximum number is the count of frames that change scene inside the video.

3.1.1 Input Secret Image

This is a secret image that we are trying to hide from a third party. At first, the secret image are splitted into parts $(p1, p2, p3, \cdots, pN)$ then applying Advanced Encryption standard AES technique for each part of image bytes to be more secured, this AES step for adding a new layer for security data [14] before embedding data into the video so that secret image will be changed to encrypted byte parts $(Ep1, Ep2, Ep3, \cdots, EpN)$. After the Encryption process is completed for all parts of secret image, it can embed each part of the image in the frames video using deep genetic algorithm approach. Deep genetic algorithm helps the proposed method for searching to select the best pixel to replace its value with the value of the secret image based on high fitness function. Fitness function is measured with the lowest difference value between image pixel of the cover image and image pixel of stego-image after applying mutation with image part value as shown in Equation (2).

$$fitness[x] = min[stegoimage[x] - coverframe[x]]. \quad (2)$$

Also, it helps for searching to select the best frame for each part of the image with respect to the highest fitness function after applying mutation on each selected frames with each part of the secret image as shown in Equation (3).

$$fitness[x] = min(coverframe[i], imagepart[j]). \quad (3)$$

The difference between the two images is saving in Pixel Index Table (pit file) and encrypted pit file with Rivest, Shamir and Adelman RSA algorithm [15] to add a level of security. We selected a genetic algorithm (GA) approach for embedding images in the video for many reasons like keeping visual video quality and more dynamically embedding to be hard for detecting with a hacker. After GA embedded successfully we recombine stego-frames together to reproduce video with secret image "stego-video".

3.1.2 Input Secret Text

This is a secret text that trying to hide from a hacker. This text is converted to bytes then encrypted using the AES technique. After the encryption process is completed, it can be embedded in the encrypted byte in stego-video (that generated from the previous step) with many methods like using Least significant bit LSB in FLV tags [16] or using the genetic algorithm to find the best pixel with high fitness. After the embedding step is completed, it will produce the final Stego-Video with encrypted images and text and can easily send the video on the public network to the receiver without a hacker suspected in.

3.2 Extracting Data Algorithm from Video

This algorithm explains how the secret image and text message will be extract from stego-video. This is an opposite technique for embedding process.

Input: stego-Video, Encryption key, Pixel Index Table (pit) file.

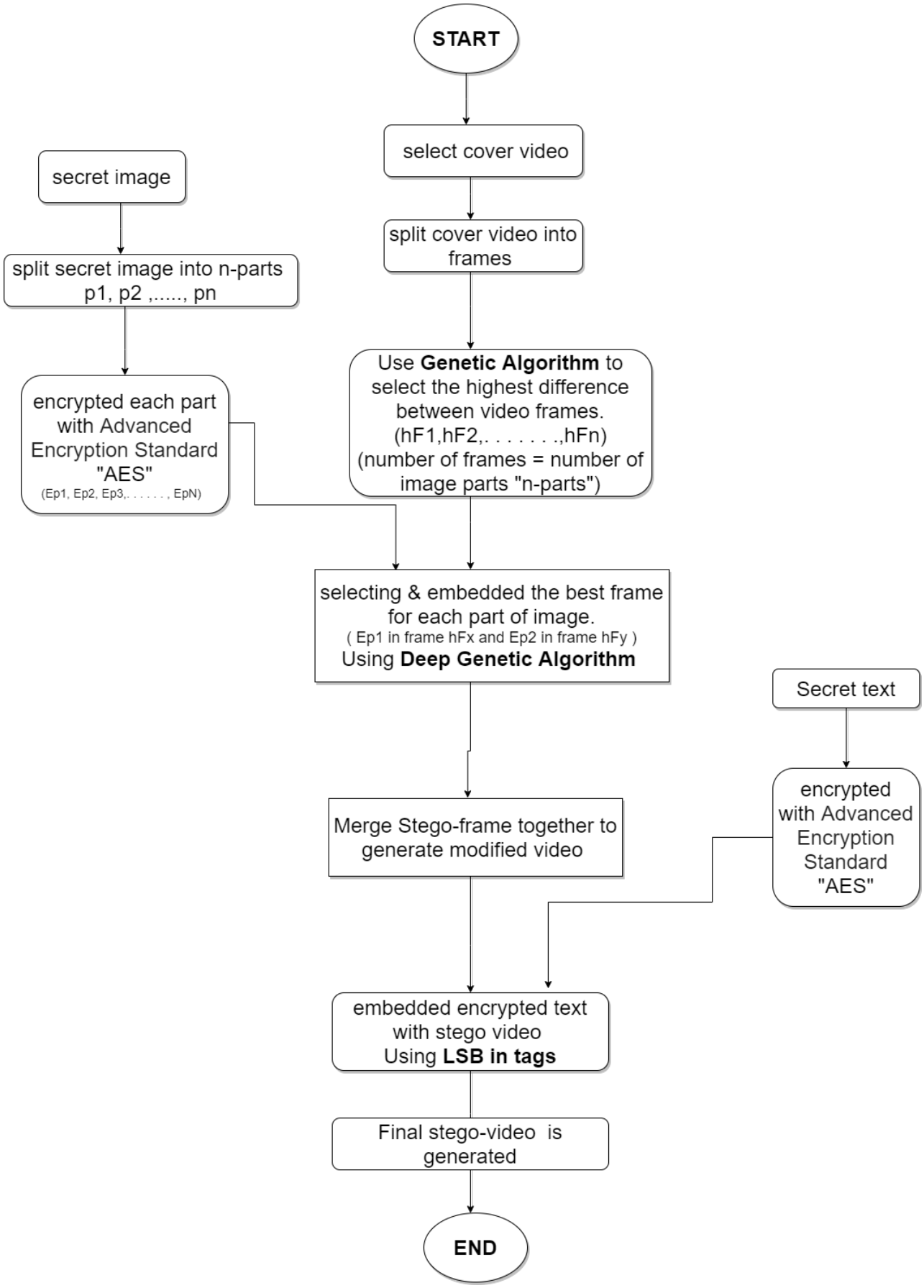


Figure 2: The proposed steganography embedding algorithm

Output: Secret Information.

Step 1: Taking Stego-video and extract secret data from the least significant bit of FLV tags and decrypted with the AES technique. From this step, the receiver has the actual message from the sender.

Step 2: Splitting stego-video into frames and selecting frames with the highest difference that sender embedded secret image parts in them.

Step 3: Decrypted pit file then use it to extract encrypted image parts from each frame.

Step 4: Decrypted image parts with AES technique.

Step 5: merge image parts to get the actual image. From this step, the receiver has the actual image from the sender.

4 Experimental Results

The experimental results discussed in this section to show and verify the performance of our proposed steganography method. We are selecting a random Flash video (.FLV extension) as cover video because FLV video can easily remove data tags [12] or add new data at the end of tags without any corruption for original video or hidden secret data at the Metadata [2] and we are using this tags for embedding secret text and for these reasons we have chosen the embedded method is Least significant bit LSB in FLV tags [16] so Adding this encrypted byte to tags is to preserve the quality of the video.

The proposed method was implemented using eclipse java version (4.12.0) and MATLAB release R2011a on Lenovo with Intel Core i7 CPU @ 2.70 GHz 2.90 GHz processors and 8 GB memory running on Microsoft Windows 10 to get the stego-video. It was found that the cover and stego-video/frames visually seemed identical. We measure the quality of the video frame by using two parameters, the first one is Mean Square Error (MSE) and the second is Peak Signal-to-Noise Ratio (PSNR) described in Equations 4 and 5. The following equations have defined these parameters:

$$MSE = \frac{1}{mn}\sum_{r=0}^{m-1}\sum_{c=0}^{n-1}[I(r,c) - K(r,c)]^2 \quad (4)$$

$$PSNR = 10.\log_{10}\frac{MAX_I^2}{MSE} \quad (5)$$

where I(r,c) is the original image and K(r,c)is the stego-image, m and n is the number of rows and columns in the input images respectively. MAXI is the maximum possible pixel value of the image. It is desirable to have low values of Mean Square Error (MSE) and high values of Peak Signal to Noise Ratio (PSNR) which gives indicator for good quality of the image and more similar to the cover image. Table 1 shows the values of PSNR and MSE of the proposed method and display histogram for the cover frame and stego-frame where the cover frame is Lena and its size is 265*265 and Figure 3 shows the secret image which is Alexandria university logo and its size is 110*143. As you can see from the result below, the histogram of the cover image and histogram of the stego-image seem approximately identical.

Figure 3: The secret image

Table 1: The PSNR, MSE and histogram Results of our proposed method

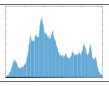
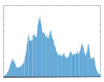
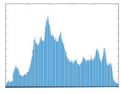
Cover image	Histogram	Secret image	Cover image after steganography	histogram	PSNR	MSE
 Lena.bmp		Top part 			42.326	3.806
 Lena.bmp		Bottom part 			44.878	2.1145

Table 2 shows the results of PSNR and MSE values for some images on the proposed method compared with Steganography Technique using the Genetic Algorithm method in [6]. They are used an advanced search algorithm such as GA to generate a key of sequence of blocks that minimizes the fitness function in which it is defined as the MSE between the original hidden text/image and the covered image. At first, initialize the population of size by rearranging the order of the blocks of the secret message using uniform random number generator. Each gene in a chromosome contains index of image pixel then using a genetic algorithm to find the optimal distribution of secret message blocks in cover. For each chromosome, the best position of each block (gene) is determined by converting each block of cover image to vector then compare all pixels of this vector with one pixel of blocks of secret message then choose minimum different and then hide the secret message/image within cover to generate the stego-image. After that, Encrypt the fit chromosome obtained from GA by adapting the BITXOR function to increase security.

But in our proposed method, At first we select the best frames using GA depending on high difference between frames and the secret image is divide it into a different numbers of parts according to its size. the secret image encrypted using AES technique and then each pixel of frame describe as chromosomes contain its position and the RGB values. we are applying cross over and mutations to get the highest fitness which is depending on the lowest difference between the video frame and the secret image

Table 2: The PSNR and MSE Results

Cover Image	Secret image	Split Secret image	Proposed Method				Steganography Method in [Essa, Abdullah, and AL-Dabbagh 2018]	
			MSE		PSNR		MSE	PSNR
London.bmp 	Three cars.bmp 	Top part 	0.2160	0.3093	54.786	53.4336	2.6083	44.0012
		Bottom part 	0.4026		52.081			
Car.bmp 	Three cars.bmp 	Top part 	0.4293	0.5877	51.802	50.602	2.1778	44.7847
		Bottom part 	0.7461		49.402			
Tree.bmp 	Three cars.bmp 	Top part 	0.0128	0.01945	67.058	65.511	0.0412	62.0193
		Bottom part 	0.0261		63.964			

so can be easily replace secret data in the selected pixel. we trying to select the best image parts that is fit in the best frame in the best pixels/positions.

It was concluded that the proposed method gave better values [lower values of MSE and higher values of PSNR] than the steganography method in [6] where the size of the secret image is 32*32 and size of the cover image is 384*384. Finally, these results were shown no changes observed between the original video and stego-video, also the size of stego-video will remain unchanged.

5 Conclusion and Future Work

The proposed steganography algorithm, which is seeking for data hiding technique, can be applied to a video to hide secret text messages and secret images inside the tags and frames of the video. The proposed algorithm gave good results because it used many levels of deep genetic algorithm. Using genetic algorithm for selecting the best frames and the best pixels to embed images with small visual distortions also using least significant bit method for hiding secret text in video tags. We conclude that the size of the video and the quality of the secret image remain the same before and after embedding. In the future, we will apply video steganography to hide a secret video using deep genetic algorithms.

References

[1] P. C. Bebe, K. Rajamani, P. Srideviponmalar and C. T. Samyuktha, "Secured implementation of steganography in multicloud," *Materials Today: Proceedings*, 2020. (`https://doi.org/10.1016/j.matpr.2020.12.900`)

[2] J. P. Cruz, N. J. Libatique, and G. Tangonan, "Steganography and data hiding in flash video (FLV)," in *IEEE Region 10 Conference*, Nov. 2012. DOI:10.1109/TENCON.2012.6412279.

[3] S. M. Darwish, S. K. Guirguis, and W. A. Alatafy, "An enhanced steganographic system for data hiding in true color images," in *The Second International Conference on Informatics Engineering & Information Science (ICIEIS'13)*, pp. 75–83, 2013.

[4] K. Dasgupta, J. K. Mondal, and P. Dutta, "Optimized video steganography using genetic algorithm (GA)," in *Procedia Technology 10 International Conference on Computational Intelligence: Modeling, Techniques and Applications (CIMTA'13)*, pp. 131–137, 2013.

[5] M. E. Eltahir, B. B. Zaidan, L. M. Kiah, and A. A. Zaidan, "High rate video streaming steganography," in *International Conference on Information Management and Engineering*, 2009. DOI: 10.1109/ICFCC.2009.44.

[6] R. J. Essa, N. A. Z. Abdullah, and R. D. AL-Dabbagh, "Steganography technique using genetic algorithm," *Iraqi Journal of Science*, vol. 59, no. 3A, pp. 1312–1325, 2018.

[7] M. M. H. Gaber, Y. F. Hassan, and K. M. Mohamed, "Cryptography with cellular automata," *International Journal of Computational and Applied Mathematics*, vol. 4, no. 1, pp. 11–18, 2009.

[8] A. E. Ibrahim, M. A. Elshahed, and T. I. Elarif, "Video steganography using least significant bit in frequency domain," *International Journal of Intelligent Computing and Information Science*, vol. 16, no. 1, pp. 89–98, 2016.

[9] M. Khodaei, B. S. Bigham, and K. Faez, "Adaptive data hiding, using pixel-value-differencing and lsb substitution," *Cybernetics and Systems*, vol. 47, no. 8, pp. 617–628, 2016.

[10] S. Limkar, A. Nemade, A. Badgujar, and R. Kate, "Improved data hiding technique based on audio and video steganography," *Smart Computing and Informatics*, vol. 78, pp. 581–588, 2018.

[11] S. Manisha and T. S. Sharmila, "A two-level secure data hiding algorithm for video steganography," *Multidimensional Systems and Signal Processing*, vol. 30, no. 2, pp. 529–542, 2018.

[12] A. J. Mozo, M. E. Obien, C. J. Rigor, D. F. Rayel, K. Chua, and G. Tangonan, "Video steganography using flash video (FLV)," in *International Instrumentation and Measurement Technology Conference*, May 2009. DOI: 10.1109/IMTC.2009.5168563.

[13] A. Pandey and J. Chopra, "Comparison of various steganography techniques using LSB and 2LSB: A

review," *International Journal of Scientic Research Engineering & Technology (IJSRET'17)*, vol. 6, no. 5, pp. 522–525, 2017.

[14] A. Pandey and J. Chopra, "Steganography using aes and LSB techniques," *International Journal of Scientific Research Engineering & Technology (IJSRET'17)*, vol. 6, no. 6, pp. 61–73, 2017.

[15] P. Patil, P. Narayankar, D. G. Narayan, and S. M. Meena, "A comprehensive evaluation of cryptographic algorithms: DES, 3DES, AES, RSA and blowfish," in *Procedia Computer Science 78 International Conference on Information Security & Privacy (ICISP'15)*, pp. 617–624, Dec. 2016.

[16] M. M. Sadek, A. S. Khalifa, and M. G. M. Mostafa, "Video steganography: A comprehensive review," in *Multimedia Tools and Applications*, vol. 74, pp. 7063–7094, 2014.

[17] U. Sahu and S Mitra, "A secure data hiding technique using video steganography," *International Journal of Computer Science & Communication Networks*, vol. 5, no. 5, pp. 348–357.

[18] D. K. Sarmah and N. Bajpai, "Proposed system for data hiding using cryptography and steganography," *International Journal of Computer Applications*, vol. 8, no. 9, pp. 7–10, 2010.

[19] K. B. Sudeepa, K. Raju, H. S. R. Kumar, and G. Aithal, "A new approach for video steganography based on randomization and parallelization," in *Procedia Computer Science 78 International Conference on Information Security & Privacy (ICISP'15)*, pp. 483–490, 2016.

[20] Y. Wang, H. Zhao, S. Liu, Y. Liu, S. Liu, "Video steganography: A review," *Neurocomputing*, vol. 335, pp. 238–250, 2019.

Biography

Nouran Mohamed Selim was born in 14 th May, 1993, Alexandria, Egypt. She obtained the B.Sc. degree in Computer Science, Faculty of Science, Alexandria University, 2015 with Grade: "Excellent with the degree of honor". Currently, she is Teaching Assistant of Faculty of Education, department of Mathematics, Alexandria University, Egypt. Her research interests include computer and information security.

Shawkat Kamal Guirguis was born in 25 th February, 1958, Alexandria, Egypt. He obtained the B.Sc. and M.Sc. Degrees in Computer Science & Automatic Control, Faculty of Engineering, Alexandria University, 1981 and 1984 respectively, with Grade: "Distinction with the degree of honor". In 1988 he obtained a Ph.D. in Electronics & Communication, Cairo University, Co-Supervised by Imperial College of Science & Technology, University of London, U.K. Currently he is Professor of Computer Science and Informatics, department of Information Technology, Institute of Graduate Studies & Research (IGSR), Alexandria University, Egypt. His current research interests include network and information security, wireless sensor networks, data mining and cloud computing.

Yasser Fouad Hassan was born in 20 th May, 1974, Alexandria, Egypt. He obtained the B.Sc. degree in Computer Science, Faculty of Science, Alexandria University, 1996. In 2003 he obtained a Ph.D. degree in Computer Engineering, Toin University of Yokohama, Japan. Currently, he is Professor of Computer Science, Faculty of computer science and AI, Pharos University, Egypt. His current research interests include Artificial Intelligence, Soft Computing, Machine Learning, Neural Networks, Cellular automata.

A Framework for Detecting Compatibility-Issues-Proneness Apps Based on Multimodal Analysis in Android Platform

Chen Xu[1], Caimei Wang[2], Yan Xiong[1], Wenchao Huang[1], Zhaoyi Meng[1], and Fuyou Miao[1]
(Corresponding author: Caimei Wang)
School of Computer Science and Technology, University of Science and Technology of China [1]
Elec-3 (Diansan) Building, West Campus of USTC, Huang Shan Road, Hefei, Anhui Province, China
Department of Computer Science, Hefei University[2]
Building 38, No. 99, Jinxiu Avenue, Hefei, Anhui Province, China
Email: wangcmo@mail.ustc.edu.cn
(Received Jan. 6, 2021; Revised and Accepted July 5, 2021; First Online Dec. 18, 2021)

Abstract

With the prosperity of Android, the compatibility issues in apps cause security flaws and bring damages to the user experience. Unfortunately, recent studies cannot help the developer to identify the apps that are compatibility-issue-proneness. It motivates us to propose an automated approach to identify these apps derived from multimodal learning. To this goal, we first present some potential modalities of apps based on previous insight and then leverage statistical methods to test the modalities. Finally, we use the selected modalities to identify these apps on a real dataset. Experimental results on apps demonstrate the effectiveness of our work.

Keywords: Android; Compatibility Issues; Modality

1 Introduction

Android has been the largest mobile platform in the world, with 74.43 % market share of global smartphone shipments in Sept 2020 [28]. For profits and the competitive power, manufacturers (*e.g.*, Xiaomi, Huawei, Samsung) choose to release new devices and customized systems on the Android platform. At the I/O developer conference in 2019, Google announced that there are more than 2.5 billion active Android devices with 180 hardware manufacturers [7]. Given the large number of devices with different hardware and system configurations, it is a non-trivial task for Android developers to ensure their apps behave as expected among those myriad devices as possible [11]. The *cross-devices inconsistencies problem* is defined as the *compatibility issue* in many studies [30,33]. The compatibility issue may bring damage to user experience and cause security issues [11,34]. For this reason, it was reported that 94% of developers identified the issues as the main reason to cause themselves to avoid working on the Android platform [17,22].

The problem of identifying compatibility-issue-proneness apps is crucial because it is good for each stakeholder in the whole Android ecosystem. For example, from the view of the market maintainer, the compatibility issues that occur in apps can be detrimental to the user base. If the problem has been solved, the mobile app market can provide maintenance advice to developers. Besides, the developer can reduce the test efforts and fix the compatibility issue for targeted apps, which is also beneficial for the user experience. The potential value of solving this problem motivates our work.

Existing researches deal with compatibility issues on Android apps from several aspects. Some studies help Android developers to prioritize major test device via user reviews [17] or app usage data [22,33]. However, the developers still need to conduct extensive testing for each app on selected test devices. Some studies discover that compatibility issues derive from multiple reasons [30], such as device variations [17,22,31], complex user interfaces [11], API evolutions [13,15,20], etc.

PIVOT [31] discovers APIs in the Android framework which are caused by compatibility issues among different devices. DiffDroid [11] leverages a differential testing to automatically identify cross-platform inconsistencies in the UI of Android apps. API-evolution-based approaches [13,20] compute the additions and removals of Android framework APIs between consecutive API levels to find fragmentation-induced compatibility issues. Nevertheless, these automated approaches can only help to detect a specific type of compatibility issues.

Detecting compatibility-issue-proneness apps is a non-trivial and difficult problem, as our goal is to find the apps that are derived from multi reason. In this paper,

Table 1: Android OS distribution

Version	Codename	API	Distribution	Release Date
4.0.3-4.0.4	Ice Cream Sandwich	15	0.2%	Dec, 2011
4.1.x	Jelly Bean	16	0.6%	Jul, 2012
4.2.x		17	0.8%	Nov, 2012
4.3		18	0.3%	Jul, 2013
4.4	KitKat	19	4.0%	Nov, 2013
5.0	Lollipop	21	1.8%	Nov, 2014
5.1		22	7.4%	Mar, 2015
6.0	Marshmallow	23	11.2%	Oct, 2015
7.0	Nougat	24	7.5%	Aug, 2016
7.1		25	5.4%	Oct, 2016
8.0	Oreo	26	7.3%	Aug, 2017
8.1	Oreo	27	14.0%	Dec, 2017
9.0	Pie	28	31.3%	Aug, 2018
10.0	10	29	8.2%	Sept, 2019

we try to solve this problem by exploiting multi-modal heterogeneous app data from different source.

The information presents conceptual characteristics of apps, and thus is helpful in addressing our problem.

The challenge in our work is how to select and use the multi modalities of data to measure the degree of compatibility-issue-proneness. The previous researches only leverage one specific type of data and detect compatibility issue derived from a single cause. To solve the problem, we model the degree of compatibility-issue-proneness by leveraging multi-modal heterogeneous data. Specifically, we select some potential modalities from different source, such as app market, resource file and class files. Then we adopt statistical approach to test whether significant difference among the two samples in term of each potential modality. In the end, we use the selected modalities to identify the compatibility-issue-proneness apps.

Our work is evaluated on a large real-world dataset consisting of 7,526 Android apps in Google Play. The evaluation results show that our work is an effective approach to identify compatibility-issue-proneness apps.

This paper makes the following contributions:

- We provide an insight into the root causes of compatibility issues in Android apps and present potential modalities of app that may influence on the compatibility issues. To the best of our knowledge, such relations have not been empirically investigated yet.
- We study the problem of identify the modeling compatibility-issue-proneness apps based on multi-modal learning.
- We conduct the evaluation on our work among a real-world dataset.

The structure of this paper is as follows. In Section 2, we present the background of compatibility issues and APK files on Android platform. We describe our modalities extraction methodology and data processing steps in Section 3. We present the experimental results of modalities extraction and app identification in Section 4. We discuss threats to validity in Section 5. Related work is described in Section 6. We finally conclude and briefly mention future directions in Section 7.

2 Background

2.1 Compatibility Issues

Compatibility issues come up when an app may not suitable for all devices that carry it. Specifically, an Android app may present different outputs across devices, and do harm to user experience [11, 33]. For example, an app behaves as expected on a Huawei device, but its behavior is inconsistent among others devices, even the app may crash on some devices.

Compatibility issues can be small, for example a features not working properly, but they can also be problematic, such as the crash of the app or the system may come up.

Compatibility issues can refer to interoperability between the device and the app. While Android devices are released frequently, it is a challenge for developers to deal with compatibility issues in apps. The developers are unable to choose some device models among thousands of devices to test compatibility issues. According to OpenSignal [26], there are more than 682,000 Android devices, covers 24,093 distinct device models and 1,294 device brands as early as 2015. OpenSignal also referred that the number is more than doubled from the 11, 868 models based on a survey in 2013. Besides, the hardware configuration composition and driver implementation of these devices varies, which bring the heavy workload to the test process.

Meanwhile, the Android OS carrying on the devices varies as well. The OSs are evolving regularly for profits and security needs, with 115 API updating per month on average [25]. Even after releasing a new version, its market could not notably increase.

Table 1 lists the information of the major Android OS versions on April 10, 2020. Note that Android Oreo had been released about 1 year, yet the market share only reached 8.2 %.

2.2 APK Files

Android Package (APK) is a package file which is used in installation and execution of an app on the Android platform. Specifically, Android apps are written in Java by leveraging Android Software Development Kit (SDK). Then, all of app's part is required to be compiled into one package file with a ".apk" extension. If the app is required to be released the app on the application market, the compilation process is necessary. With the APK file, user can manually install the app on the Android device.

For the intellectual property right, the resource code of apps is not available to the public. The primacy approach to analyze the Android app is to transform Java bytecode into intermediate representation for code analysis [3, 19].

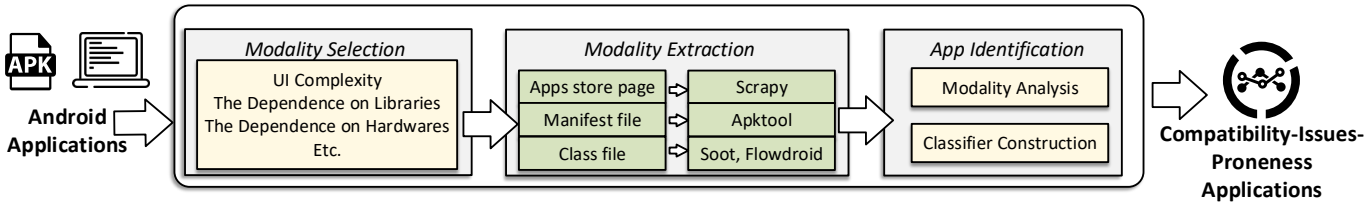


Figure 1: Architecture of our work

3 Empirical Study Methodology

Figure 1 illustrates the workflow of our work. first, we select the potential modalities based on root causes of compatibility issues in apps. Then we extract these modalities from three sources that are store page, manifest file and class file. For the each modality, we measure its usability in app identification, and finally build a classifier to identify the compatibility-issues-proneness apps.

3.1 Modality Selection

First we choose 5 potential modalities and elaborate them as follows:

The category. Indicates a group of apps having appropriately similar characteristics on the application market. It also presents the purpose of this type of app for users to select. For example, the apps, with the Entertainment category tagging, are often used to fill your time and enjoy yourself. Developers might pay varies attention to the compatibility issues among different categories of apps. For example, the developer might be more conscious about compatibility issues for financial apps than other apps. Till now, there are 32 app categories in the Google Play Store.

The size. Indicates the size of code in apps, which includes the size of the APK file and the LOC in the decompiled Java files. The larger app in terms of amounts of code may be harder to be maintained. As a result, large size code in apps may introduce more compatibility issues in apps.

The UI complexity. Indicates the amount of the user interface in apps. Note that activity represents a single-screen UI in apps. A representation of compatibility issues is the inconsistent output on the UIs. Therefore, the app with more complex UIs may have more compatibility issues. Here we use the amount of activity to measure the UI complexity.

The dependence on libraries. Indicates the amount of the dependence of the third-party libraries in apps. The third-party libraries are extensively used in Android to provide functionalities and ease human cost in the development process. However, the dependence on these libraries may bring compatibility issues in apps.

The dependence on hardwares. Indicates the amount of the dependence of hardwares in apps. Some functions in apps rely on hardware composition. For example, A social app have to use the camera to provide features with photos. However, compatibility issues may occur with the problematic implementations of hardware drivers.

3.2 Modality Extraction

In our work, the modalities are derived from three sources as follows.

Google play store page. The description of the app on the store page explicitly presents information (*e.g.*, descriptions, category, install size, user reviews, installs and rates) to users. Therefore, the information can be achieved by indirectly processing the google play store page [24].

Manifest file. The AndroidManifest.xml file is located in the root directory of project source set, which is used to statically define some essential information of the app.

For each component (*e.g.*, activity, service) that the developer creates, and each permission or hardware feature that app requires, must be declared in it [4]. For example, if an app needs to access the camera, CAMERA permission is necessary to be declared in its AndroidManifest.xml. Specifically, the manifest file would have *<uses-permission android:name="android.permission.CAMERA">* in it. Besides, to relieve compatibility issues, the developer can declare the minimum API level required to run the app in AndroidManifest.xml. Note that each line in the manifest file always begins with an element (*e.g.*<activity>, <service>, <uses-permission>), which is used to indicate the kind of the information in the line.

Class files. Some modalities involving code complexity required static code analysis. We first decompile the Android apps from their DEX byte code into intermediate Smali code by Apktool [2]. To capture the dependency of libraries, we use package names in the Smali code to identify third-party libraries. The same process is also used by recent studies [5, 23].

3.3 Dataset

The dataset is built on a server with an Intel Xeon E5-2620V4 2.2G CPU and 128 GB physical memory. We first randomly collect 30,000 apps in Androzoo [1]. Note that these apps were derived from Google Play. We map each app in our dataset with its Google Play store page if exists and leverage the scrapy framework [27] to extract the data. Then, we use Apktool [2] to decompile the APK file and extract AndroidManifest.xml, and leverage FlowDroid [3] to conduct code analysis. Specifically, we remove some subject apps out of 30,000 apps for some reasons below:

- **Unavailable store page.** Some apps maybe withdrawn by developers or pulled off shelves by market maintainers thus failed to be found in Google Play.
- **Low installs.** We filter apps if they have less than 100 installs to make sure the quality of the apps.
- **Limitations of the code analysis tool.** Some apps are heavily obfuscated and unable to be used in our evaluation. Besides, FlowDroid [3] runs out of memory or exceeded the time limit or threw the Soot exception in the process of doing static analysis on some apps.

After the filtering process, we have 7,526 apps in the dataset. We present a summary of these apps with the descriptive statistics in Table 2.

In order to measure the degree of compatibility-issue-proneness in term of each modality, we compute the statistical differences among these apps whose amount of compatibility issues differs. Since no existing the ground truth to distinguish the app has compatibility issues or not (it is also our motivation), we conduct program analysis on these apps and manually label each app is compatibility-issues-proneness or not in term of the compatibility-related APIs it contains. We leverage the statistical result of CRA provided by the tool ICARUS [32] and Pivot [31].

Besides, note that third-party libraries account for a large portion of the code in Android apps, program analysis on Android apps typically requires detecting or removing third-party libraries first. We remove the codes imported by third-party libraries via a library list. More precisely, we first add the package names of the identified libraries into a list and remove such packages according to the list in decompiled apps. The test result shows that about over 95 % apps in our dataset contain the third party libraries such as com/google/ads, com/-facebook and com/umeng. By program analysis, we sort each app into three samples in term of the amount of compatibility-related APIs it contains, and then consider the bottom 10% apps as the high-quality apps and top 10% apps as compatibility-issues-proneness apps. Finally, we get 139 high-quality apps and 131 compatibility-issues-proneness apps for the study.

Table 2: Summary of the Apps Used in Our Dataset

Category	Apps(%)	KLOC
Arcade	327 (4.3%)	623-8K
Books and reference	416 (5.5%)	92-18K
Brain	357 (4.7%)	1K-32K
Business	257 (3.4%)	1K-18K
Casual	426 (5.7%)	372-12K
Comics	14 (0.2%)	1K-2K
Communication	610 (8.1%)	297-12K
Education	307 (4.1%)	1K-8K
Entertainment	825 (11.0%)	493-8K
Finance	142 (1.9%)	726-14K
Games	294 (3.9%)	79-146K
Health and fitness	21 (0.3%)	1K-28K
Libraries and demo	124 (1.6%)	182-7K
Lifestyle	214 (2.8%)	2K-7K
Media and video Cards	610 (8.1%)	1K-23K
Medial	12 (0.2%)	2K-8K
Music and audio	119 (1.6%)	1K-11K
News and magazines	163 (2.2%)	615-17K
Personalization	491 (6.5%)	393-8K
Photography	316 (4.2%)	194-87K
Productivity	241 (3.2%)	782-39K
Racing	121 (1.6%)	4K-38K
Shopping	73 (0.1%)	1K-5K
Social	92 (1.2%)	2K-13K
Sports	243 (3.2%)	870-43K
Tools	619 (8.2%)	169-38K
Transportation	10 (0.1%)	3K-13K
Travel and Local	71 (0.9%)	3K-8K
Weather	11 (0.1%)	708-7K
Total	7526	109-146K

4 Study Results

This section presents and discusses the results of our selected modalities in Section 3.1. First, we leverage the statistical methods to measure the association between each modality and compatibility issues in apps. Then we use these modalities to identify the compatibility-issues-proneness apps.

4.1 Evaluation on Modalities

Approach. Here, we measure each selected modalities among two group of samples. Specifically, we first analyze the statistical significance of the difference between the two samples that respectively contain 139 high-quality apps and 131 compatibility-issues-proneness apps by applying non-parametric Mann-Whitney-Wilcoxon (MWW) test [10] at p-value =0.01 [8]. We also used Cliff's Delta statistic that is a nonparametric effect size to measure effect size of the difference between the two groups [12].

We interpret the effect size values as small for 0.147 <d <0.33, medium for 0.33 <d <0.474, and large for d >0.474 with the guidelines in previous work [14,21, 29].

Results. We find that the two groups have statistically significant differences in term of selected modalities except for size and category.

In terms of size and category, we found no signifi-

cant difference with d of 0.116 and 0.071. For UI complexity, the effect size is medium with d of 0.372.

In terms of dependence of libraries and hardwares, the results show statistical significant difference, with p-value<of 0.0001 and the large effect size (d=0.682, 0.517). We discuss each modality one by one as follows.

UI complexity. Compared with high-quality apps, compatibility-issues-proneness apps tend to have more complexity UI. In fact, a symptom of compatibility issues is the inconsistencies in the UI of Android apps among various devices. Users have direct interaction with the UIs of the apps thus the inconsistent behaviors among UIs are easily noticed. When an app has more UIs to present to the user, it is more difficult to maintain app compatibility. For example, an app may not have specific UI declarations for diverse density screens.

Besides we find the APIs which directly involving what is displayed on the device screen are prone to compatibility issues. Such as the class *android.widget.ZoomButtonsController*, which is used to handle showing and hiding the zoom controls and positioning it relative to an owner view in UI. However, the class was deprecated in API level 26 and may introduce compatibility issues on devices carrying the new OS. If the app contains more UI, the developers may unconsciously use these APIs thus bring compatibility issues.

Dependence on libraries. Compared with high-quality apps, compatibility-issues-proneness apps tend to have more dependence on libraries. Although the usage of libraries eases the development process, the third-party code of the libraries introduce more compatibility issues. For some libraries, the potential compatibility it may introduce may not explicitly indicate in its documentation.

To elaborate our results, we present a code segment in keepass2android [16], a popular Android project with 11,410 stars on GitHub, as shown in Listing 1. In this example, the app leverages a library to create a cipher object for encryption and decryption. Before invoking the library, the developer probes the device model, and verifies whether it is Acer Iconia A500 (Line 2) and records the result into a boolean value. The boolean value is then used in the conditional statement (Line 7) to control the callsite of API *Cipher.getInstance()*. Notably, the developer writes "The Acer Iconia A500 is special and seems to always crash in the native crypto libraries" as an annotation (not showing on the code example for brevity) below this line, which implies that the usage of the API provide by the library may cause the compatibility issue.

```
1 public static boolean deviceBlacklisted() {
2   blacklisted = Build.MODEL.equals("A500");
3   return blacklisted;
4 }
5
6 public static Cipher getInstance( ) {
7   if (!deviceBlacklisted())
8     return Cipher.getInstance();
9   }
```

Listing 1: A code segment in keepass2android

As the example shows, the dependence on libraries may cause serious compatibility issue. Therefore the documents of third-party libraries are required to be carefully read. However, most libraries either lack a full documentation or do not indicate its potential compatibility issue in the documents.

Dependence on hardwares. Compared with high-quality apps, compatibility-issues-proneness apps tend to have more dependence on hardwares. The more hardwares the app used, the greater chance for compatibility issues arise. The function of hardwares relies on low-level drivers, whose implementations can make inconsistent behaviors among different devices. Besides, the diversity of hardware composition can easily lead to compatibility issues.

For example, the usage of SD card may introduce the compatibility issues. Some devices (*e.g.*, Samsung Galaxy S2, HTC Evo 4G) do not use the external storage convention /mnt/sdcard/. Besides, there exists other devices with no SD card and multiple SD cards on the market. To deal with the issues, developers have to make extra effort among the various devices. Specifically, the developers may hardcode the SD card path for some targeted devices. However, the issues would still occur since the new devices continue to emerge.

4.2 Evaluation on Identification

Approach. We leverage the 3 modalities to identify the compatibility-issues-proneness apps. We use the labeled apps both as training and test data in a ten-fold cross-validation [18], which is a standard approach for evaluating the approach. Specifically, we partition the apps in 10 subsets, and we use 9 subsets for training the model and 1 for testing. We run this for 10 times, each time we use a different subset for testing. Here we adopt the support vector machine (SVM) as the classifier.

Metrics. We consider two evaluation metrics, the precision and recall. Precision means the fraction of compatibility-issues-proneness apps correctly identified as compatibility-issues-proneness apps among those labeled. Recall means the fraction of compatibility-issues-proneness apps correctly identified as compatibility-issues-proneness apps among those reported by our approach.

Given the ground truth and the detection results, there are four possible outcomes: True positive(TP),

Table 3: Effectiveness of our approach to identify compatibility-issues-proneness apps

Modalities	Recall(%)	Precision(%)
The UI Complexity (UC)	72.1	74.1
The Dependence on Libraries (DL)	78.2	70.1
The Dependence on Hardwares (DH)	75.1	72.2
UC & DH	75.5	76.3
UC & DL	85.2	80.5
DL & DH	82.3	76.4
Total (UC & DL & DH)	84.3	80.4

true negative (TN), false positive (FP) and false negative (FN). TP means that an app is compatibility-issues-proneness with respect to the ground truth and it is identified by our approach. TN means that an app is compatibility-issues-proneness with respect to the ground truth and our approach does not identify it correctly. FP means that an app is not compatibility-issues-proneness with respect to the ground truth but our approach identifies it by mistake. FN means that an app is not compatibility-issues-proneness with respect to the ground truth but our approach truly does not identify it. Finally, the precision and recall are computed by the following formulas:

$$Precison = \frac{TP}{TP+FP}$$

$$Recall = \frac{TP}{TP+FN}$$

Finally, we report the average precision and recall in Table 3.

Results. Results show that our work achieves both higher recall and precision. The precisions and the recalls of the analysis results for apps from different categories are listed in Table 3 (the last two columns). The first column shows the modalities used. The second and the third column list the recall and precision. We also present the experimental results based on each single modality and the combination of different modalities. Note that each modality is useful to improve both the precision and recall of the identification results.

We also analyze the misidentified samples. Specifically, we invite three Android experts to test the apps that are misidentified by our approach. We find some apps that are false positives have compatibility issues in the practical scenario. However, due to the limitation of the compatibility-related APIs dataset, we label them as negative samples. To improve the recall and precision of identification, we will label the app with consideration for test results from Android devices in the future.

5 Threats to Validity

In this section, we present and discusses threats to validity as follows.

Construct validity. Is related to whether our study reflects real-world situations. A possible threat to the validity of our study could be due to the limitation of the dataset. In our study, we have tried our best to make dataset general and representative. Given the fact that the Android platform and app ecosystems are quickly evolving, the investigated apps in dataset over five years, which make sure our experimental results may generalize to most apps.

Internal validity. Is related to uncontrolled aspects that may affect the experimental results. Our results are based on the static analysis that may be subject to issue from analysis tool Apktool and Flowdroid. We may consciously or unconsciously favor the results it presents. Another threat is related to the manual inspection in misidentified samples. We indeed understand that such manual inspection can be error-prone, so this activity has been done with special attention, double-checking and support of the second and third experience developers. We believe that the threat to construct validity is minimal.

External validity. Is related to the possibility to generalize our results. We try to study several apps from different categories. Note that a threat to external validity is that we focus on the free apps in Google Play rather than the paid apps whose APK files are difficult to collect. To be fully conclusive, we will construct our study with paid apps in the future. The apps on platforms other than Android are outside the scope of this paper.

6 Related Work

Some of the existing researches are confined to help developers to find compatibility issue in development test. Lu *et al.* [22] mined large-scale usage data from Wandoujia, and proposed an approach to prioritizing Android device models for individual apps to help developers to identify compatibility issues, based on mining large-scale usage data from Wandoujia. Khalid *et al.* [17] also helped game app developers deal with a similar problem. They picked the devices that have the most impact on app ratings by studying the reviews of game apps. Zhang *et al.* [33] proposed a systematic and cost-effective mobile compatibility test method for selecting mobile devices and their diverse platforms and configurations. Mattia *et al.* [11] automatically identify cross-platform inconsistencies in the UI of Android apps. These proposed schemes have effectively helped developers identify whether compatibility issues are occurring in the test process, but we note that it is challenging for developers to deal with compatibility issue in code-level. Since that existing work dis-

covered that developers are unable to resolve nearly 40% reported crashes [9], which is a possible consequence of the compatibility issue [17, 20, 30, 34].

Besides, some studies are proposed to understand the compatibility issue at code-level in Android apps. Wei *et al.* [30] conducted the first empirical study of compatibility issues caused by Android fragmentation in real-world Android apps at the source code level. Specifically, their work manually studied root causes, symptoms and fixing strategies of compatibility issues in open-source apps. Cai *et al.* [6] conducted a large-scale study of app compatibility issues in Android, concerning the occurrences of these issues at installation time and runtime. Specifically, their work gathered the app trace as well as the system log on the apps' executions and installations, and then analyzed these logs to recognize the execution and installation as a success or failure with related reasons. FicFinder [30] is used to automatically detect compatibility issues in Android apps, but its performance completely relies on the investigation into open-source apps, which requires the labor-intensive process and may lead to a high rate of false negatives. CiD [20] generalizes FicFinder to more compatibility issues, with mining of Android framework versions and modeling the lifecycle of all API methods. PIVOT [31] extracts and prioritizes API-device correlations from a given corpus of Android apps, and consider APIs in such correlations are compatibility issues derived from the device causes. However, they cannot provide deeper insights to help the developers to relieve the human effort in the development process. one can possibly assume that some modalities in apps may jeopardize its compatibility, to the best of our knowledge such relations have not been empirically investigated yet. Our work shed light on the relationship between the compatibility issues and some modalities of Android apps, which is a complement to recent studies.

7 Conclusion

We have contributed to this paper with a novel approach to identify compatibility-issues-proneness apps. We also present some modalities of apps that are related to compatibility issues in Android apps. To this goal, our approach starts with the analysis of given apps and extracts the potential modalities from Google Play and APK files. We then use a statistical approach to measure the association and leverage a classifier to identify the compatibility-issues-proneness apps. The evaluation on a large real-world dataset shows that the accuracy and validity of these modalities.

Acknowledgments

We thank the anonymous reviewers for their valuable comments. The research is supported by the National Key R&D Program of China 2018YFB2100300, 2018YFB0803400, National Natural Science Foundation of China under Grant No.61972369, No.61572453, No.61520106007, No.61572454, and the Fundamental Research Funds for the Central Universities, No. WK2150110009.

References

[1] K. Allix, T. F. Bissyandé, J. Klein, and Y. L. Traon, "Androzoo: Collecting millions of android apps for the research community," in *IEEE/ACM 13th Working Conference on Mining Software Repositories (MSR'16)*, pp. 468–471, 2016.

[2] Apktool, *A Tool for Reverse Engineering Android apk Files*, July 6, 2021. (`https://ibotpeaches.github.io/Apktool/`)

[3] S. Arzt, S. Rasthofer, C. Fritz, E. Bodden, A. Bartel, J. Klein, Y. L. Traon, D. Octeau, and P. McDaniel, "FlowDroid: Precise context, flow, field, object-sensitive and lifecycle-aware taint analysis for android apps," *ACM SIGPLAN Notices*, vol. 49, pp. 259–269. ACM New York, NY, USA, 2014.

[4] K. W. Y. Au, Y. F. Zhou, Z. Huang, and D. Lie, "PScout: Analyzing the android permission specification," in *Proceedings of ACM Conference on Computer and Communications Security*, pp. 217–228, 2012.

[5] M. Backes, S. Bugiel, and E. Derr, "Reliable third-party library detection in android and its security applications," in *Proceedings of ACM SIGSAC Conference on Computer and Communications Security*, pp. 356–367, 2016.

[6] H. Cai, Z. Zhang, L. Li, and X. Fu, "A large-scale study of application incompatibilities in android," in *Proceedings of the 28th ACM SIGSOFT International Symposium on Software Testing and Analysis*, pp. 216–227, 2019.

[7] J. Callaham ,*Android Now Running on Over 2.5 Billion Active Hardware Devices*, 2020. (`https://www.androidauthority.com/android-2-5-billion-devices-983534/`)

[8] W. J. Conover, *Practical Nonparametric Statistics*, vol. 350, 1998. (`http://140.117.153.69/ctdr/files/857_1734.pdf`)

[9] L. Fan, T. Su, S. Chen, G. Meng, Y. Liu, L. Xu, G. Pu, and Z. Su, "Large-scale analysis of framework-specific exceptions in android apps," in *IEEE/ACM 40th International Conference on Software Engineering (ICSE'18)*, pp. 408–419, 2018.

[10] M. P. Fay and M. A. Proschan, *Wilcoxon-Mann-Whitney or T-test? On Assumptions for Hypothesis Tests and Multiple Interpretations of Decision Rules*, vol. 4, pp. 1. , 2010.

[11] M. Fazzini and A. Orso, "*Automated Cross-Platform Inconsistency Detection for Mobile Apps*," vol. 4, pp. 308–318, 2017.

[12] R. J. Grissom and J. J. Kim, *Effect Sizes for Research: A Broad Practical Approach*, 2005. (https://psycnet.apa.org/record/2005-04135-000)

[13] D. He, L. Li, L. Wang, H. Zheng, G. Li, and J. Xue, "Understanding and detecting evolution-induced compatibility issues in android apps," in *Proceedings of the 33rd ACM/IEEE International Conference on Automated Software Engineering*, pp. 167–177, 2018.

[14] A. Hora, M. T. Valente, R. Robbes, and N. Anquetil, "When should internal interfaces be promoted to public?," in *Proceedings of the 24th ACM SIGSOFT International Symposium on Foundations of Software Engineering*, pp. 278–289, 2016.

[15] H. Huang, L. Wei, Y. Liu, and S. C. Cheung, "Understanding and detecting callback compatibility issues for android applications," in *Proceedings of the 33rd ACM/IEEE International Conference on Automated Software Engineering*, pp. 532–542, 2018.

[16] keepass2android, *Password Manager App for Android*, July 6, 2021. (https://github.com/PhilippC/keepass2android/)

[17] H. Khalid, M. Nagappan, E. Shihab, and A. E. Hassan, "Prioritizing the devices to test your app on: A case study of android game apps," in *Proceedings of the 22nd ACM SIGSOFT International Symposium on Foundations of Software Engineering*, pp. 610–620, 2014.

[18] R. Kohavi, *et al.*, "A study of cross-validation and bootstrap for accuracy estimation and model selection," in *Proceedings of the 14th International Joint Conference on Artificial Intelligence (IJCAI'95)*, vol. 14, pp. 1137–1145, 1995.

[19] P. Lam, E. Bodden, O. Lhoták, and L. Hendren, "The soot framework for java program analysis: a retrospective," in *Cetus Users and Compiler Infastructure Workshop (CETUS'11)*, vol. 15, pp. 35, 2011.

[20] L. Li, T. F. Bissyandé, H. Wang, and J. Klein, "CiD: Automating the detection of api-related compatibility issues in android apps," in *Proceedings of the 27th ACM SIGSOFT International Symposium on Software Testing and Analysis*, pp. 153–163, 2018.

[21] M. Linares-Vásquez, G. Bavota, C. Bernal-Cárdenas, M. D. Penta, R. Oliveto, and D. Poshyvanyk, "API change and fault proneness: A threat to the success of android apps," in *Proceedings of the 9th Joint Meeting on Foundations of Software Engineering*, pp. 477–487, 2013.

[22] X. Lu, X. Liu, H. Li, T. Xie, Q. Mei, D. Hao, G. Huang, and F. Feng, "Prada: Prioritizing android devices for apps by mining large-scale usage data," in *IEEE/ACM 38th International Conference on Software Engineering (ICSE'16)*, pp. 3–13, 2016.

[23] Z. Ma, H. Wang, Y. Guo, and X. Chen, "Libradar: Fast and accurate detection of third-party libraries in android apps," in *Proceedings of the 38th International Conference on Software Engineering Companion*, pp. 653–656, 2016.

[24] W. Martin, F. Sarro, and M. Harman, "Causal impact analysis for app releases in google play," in *Proceedings of the 24th ACM SIGSOFT International Symposium on Foundations of Software Engineering*, pp. 435–446, 2016.

[25] T. McDonnell, B. Ray, and M. Kim, "An empirical study of API stability and adoption in the android ecosystem," in *IEEE International Conference on Software Maintenance (ICSM'13)*, pp. 70–79, 2013.

[26] Opensignal, *Android Fragmentation Visualized*, Aug. 2015. (https://www.opensignal.com/sites/opensignal-com/files/data/reports/global/data-2015-08/2015_08_fragmentation_report.pdf)

[27] Scrapy, *An Open Source and Collaborative Framework for Extracting the Data*, July 6, 2021. (https://scrapy.org/)

[28] Statcounter, *Mobile Operating System Market Share Worldwide*, 2020. (https://gs.statcounter.com/os-market-share/mobile/worldwide/)

[29] I. Steinmacher, G. Pinto, I. S. Wiese, and M. A. Gerosa, "Almost there: A study on quasi-contributors in open-source software projects," in *IEEE/ACM 40th International Conference on Software Engineering (ICSE'18)*, pp. 256–266, 2018.

[30] L. Wei, Y. Liu, and S. C. Cheung, "Taming android fragmentation: Characterizing and detecting compatibility issues for android apps," in *Proceedings of the 31st IEEE/ACM International Conference on Automated Software Engineering*, pp. 226–237, 2016.

[31] L. Wei, Y. Liu, and S. C. Cheung, "Pivot: Learning API-device correlations to facilitate android compatibility issue detection," in *IEEE/ACM 41st International Conference on Software Engineering (ICSE'19)*, pp. 878–888, 2019.

[32] C. Xu, Y. Xiong, W. Huang, Z. Meng, F. Miao, C. Su, and G. Mo, "Identifying compatibility-related apis by exploring biased distribution in android apps," in *IEEE/ACM 42nd International Conference on Software Engineering: Companion Proceedings*, pp. 280–281, 2020.

[33] T. Zhang, J. Gao, J. Cheng, and T. Uehara, "Compatibility Testing Service for mobile applications," in *IEEE Symposium on Service-Oriented System Engineering*, pp. 179–186, 2015.

[34] X. Zhou, Y. Lee, N. Zhang, M. Naveed, and X. Wang, "The peril of fragmentation: Security hazards in android device driver customizations," in *IEEE Symposium on Security and Privacy*, pp. 409–423, 2014.

Biography

Chen Xu received the B.S. degree in computer science from Anhui Agricultural University in 2014. He is currently working towards the Ph.D. degree at the Department of Computer Science and Technology, University of Science and Technology of China. His current

research interests include information security and software engineering. (Email: kakyo82@mail.ustc.edu.cn)

Caimei Wang received the Ph.D degree in computer science from University of Science and Technology of China in 2018. She is an associate professor in School of Artificial Intelligence and Big Data, Hefei University. Her main research interests include computer network, trusted computing and information security. (Email: wangcmo@mail.ustc.edu.cn)

Wenchao Huang received B.S. and Ph.D. degrees from University of Science and Technology of China in 2006 and 2011 respectively. He is an associate professor currently with Department 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. (Email: huangwc@ustc.edu.cn)

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. (Email: yxiong@ustc.edu.cn)

Zhaoyi Meng received the B.S. degree in information security from University of Electronic Science and Technology of China in 2014, and the Ph.D. degree in computer science and technology from University of Science and Technology of China. He is currently a Post-Doctoral Researcher with the Department of Computer Science and Technology, University of Science and Technology of China. His current research interests include Android security and software formal verification. (Email:mzy516@ustc.edu.cn)

Fuyou Miao received his Ph.D of computer science from University of Science and Technology of China in 2005. He is an associate professor in the School of Computer Science and Technology, University of Science and Technology of China. His research interests include information security, information coding key management in WSN, and network security. (Email:mfy@ustc.edu.cn)

Network Security Risk Assessment Based on Enterprise Environment Characteristics

Yunxue Yang[1], Zhenqi Yang[2], Qin Yang[1], Guohua Ji[1], and Shengjun Xue[1]
(Corresponding author: Yunxue Yang)

Department of Computer Science and Technology, Silicon Lake College[1]
168 Greenland Avenue, Huaqiao International Business Zone, Kunshan, Jiangsu, China
Binjiang College of Nanjing University of Information Science & Technology, China[2]
333 Xishan Avenue, Wuxi, Jiangsu, China
Email: brightyyx@qq.com
(Received Jan. 15, 2021; Revised and Accepted June 6, 2021; First Online Dec. 18, 2021)

Abstract

The authors studied the issue of network security risk assessment and proposed a method for the network security risk assessment based on enterprise environment. First of all, the authors proposed a vulnerability severity risk assessment method based on economic losses of an enterprise to evaluate the vulnerability severity for the enterprise. Next, the authors proposed a dynamic security risk assessment method by using the Bayesian attack graph model and combining the changes of network environment. Last, the case study interpreted the detailed calculation processes of the dynamic security risk assessment method, and the simulation experiment showed that the proposed method conforms to the real threat level of the network or information system evaluated, therefore, the evaluation results are more accurate and objective.

Keywords: Bayesian Attack Graph; Bayesian Inference; Dynamic Risk Assessment; Network Security

1 Introduction

With the rapid development of computer technology and network technology, the application of computers and networks has penetrated into all aspects of social life. However, due to security vulnerabilities in network systems, the number and variety of network attacks have multiplied, making network security problems more and more serious [8]. Typical cases include: In June 2019, Canada's largest credit cooperative, Desjardins, encountered security breaches and 2.9 million customer information was leaked. In 2020, many websites such as China's JD.com could not be accessed normally due to man-in-the-middle attacks, resulting in large-scale network hijacking incidents. The attack is likely to be based on the DNS system or at the operator level. Currently, users in some areas are mainly affected by all operators. For example, China Mobile, China Unicom, China Telecom and Education Network can reproduce the hijacking problem [12, 18].

Network security incidents have caused huge economic losses to enterprises. According to Allianz Risk Barometer Top Business Risks 2020 issued by Allianz Global Corporate & Specialty Risk (AGCS) [1], cybercrime caused a global economic loss of $1.5 trillion, of which about 50% occurred in the top 10 economies of the world and this loss is expected to hit $2.5 trillion this year with an increase of 60%.

In order to solve network security problems and conduct security management and control, network security risk assessment has become a research hotspot in the field of information security. The results of the security risk assessment not only reflect the security status of the network or information system, but also predict the possibility of future attacks on the network and risks brought by these possible attacks. This is the main basis for security administrators to take further security risk control measures. The current cybersecurity risk assessment methods mainly use the models such as attack tree, attack graph and Petri net to model network attacks and analyze various possible attacks and the relationship between them [14]. These models mainly quantitatively evaluate the attack probabilities of network nodes from the perspective of the vulnerabilities existing in the network and the associated utilization of the vulnerabilities.

An important aspect of cybersecurity risk assessment is the assessment of security vulnerabilities that exist in the network. An effective assessment of security vulnerabilities can improve the effectiveness of patches and system security hardening and the typical example of this aspect is the Common Vulnerability Scoring System (CVSS) [4]. CVSS is a vulnerability assessment standard jointly issued by the US Information Security Response and Security Group and the General Security Vulnerability Scoring System Expert Group in 2007. The current common standard of CVSS is version 3.1 published in 2019 [5].

CVSS uses quantitative score to determine the risk level of vulnerability from a technical perspective. In some publicly available vulnerability databases and scanning tools, CVSS method is commonly used. CVSS evaluates the risk of a vulnerability through three measure groups: The base measure group, the time measure group, and the environment measure group. However, in the actual situation, usually only the basic metric group is used, and the time metric group and the environmental metric group are not universally applicable [21].

Since the environmental characteristics of the enterprise affected by the vulnerability are not considered, the same vulnerability risk score is often calculated by using CVSS in the different enterprise environments. However, the impact of vulnerabilities on various corporate organizations is very different in the real world. Some previous research work has also raised this issue and it is recommended to use the CVSS method carefully to determine the risk of vulnerabilities [6]. Moreover, technically dangerous vulnerabilities do not necessarily have a large economic impact on corporate institutions, which is not uncommon [11]. Current cybersecurity risk assessment methods, such as the attack tree and attack graph models are based on a risk assessment of security vulnerabilities that exist in the network. However, the shortcoming of the current work is that when calculating the risk of a node (the probability of an attacker reaching the node), it only uses the CVSS base score of the vulnerability ignoring the characteristics of the vulnerability in a specific enterprise environment, such as the confidentiality, integrity and availability requirements of the enterprise, as well as the economic losses caused by the vulnerability. Because the risk assessment of the vulnerability is inaccurate, it is impossible to obtain an accurate cybersecurity risk assessment result that is consistent with the actual situation of the enterprise [9].

In summary, in order to develop a reliable cybersecurity risk assessment method that is consistent with the actual situation of the enterprise, it is necessary to fully consider the environmental background information of a specific enterprise. First, the risk of the vulnerability should be assessed based on the characteristics of the enterprise environment, and then the network security risk assessment should be conducted within the enterprise. Based on the above observations, the main contributions of this paper are:

1) In order to assess the risk of security vulnerabilities, this paper proposes a set of metrics based on the economic loss of enterprises.

2) In order to assess the risk of security vulnerabilities quantitatively, this paper proposes a quantitative method to integrate CVSS metrics, enterprise's economic loss metrics and enterprise's security requirements metrics.

3) Based on the above two points, this paper proposes a dynamic security risk assessment method (NSRAEE), which can be combined with the environmental changes of enterprises to assess the enterprises' risks dynamically.

2 Related Work

2.1 Security Vulnerability Assessment

In order to assess the seriousness of system vulnerabilities, Karie *et al.* [9] proposed a quantitative evaluation model based on grey evaluation method and analytic hierarchy process. Mahdavifar *et al.* [11] selected the access route, and used the complexity and degree of influence as the three elements to evaluate the threat of the vulnerability. The users used the analytic hierarchy process to establish the evaluation model and the vulnerability level of the vulnerability was classified as super-risk, high-risk, medium-risk and low risk. Atapattu *et al.* [2] used a medical "case-control study" approach to compare the severity and availability of vulnerabilities. Xiao *et al.* [20] used fuzzy analytic hierarchy approach to evaluate the security level of software vulnerabilities, and further considered human subjectivity in reality, emphasized the relationship between different factors affecting information security, improved the traditional fuzzy comprehensive decision model, and proposed fuzzy integral decision model.

Zhu *et al.* [23] proposed a new vulnerability rating and scoring system (VRSS) based on the existing vulnerability level system. VRSS combines the advantages of the existing vulnerability level system and can qualitatively determine vulnerability threat levels and rate vulnerabilities quantitatively. In order to further improve the quality of vulnerability scores, Rosli *et al.* [7,13] used the analytic hierarchy process to classify vulnerabilities through vulnerability types and quantitatively describe the characteristics of vulnerability types on the basis of VRSS, thus improving the quality of vulnerability scores.

2.2 Cybersecurity Risk Assessment

The traditional cybersecurity risk assessment methods mainly use the models of attack tree, attack graph and Petri net to model network attacks and analyze various possible attacks and the relationship between them. These models mainly quantitatively evaluate the attack probability of network nodes from the point of view of vulnerabilities existing in the network and the correlation of vulnerabilities.

In order to further study the uncertainties in cyberattacks, some probabilistic models are proposed to study the quantitative assessment of cybersecurity risks, including Markov decision process models, Bayesian networks, Bayesian attack graphs and other models. These approaches model the uncertainties in the existence of cyber attacks. For example, Wang *et al.* [3] proposed a probabilistic model for assessing cybersecurity risks, using attack graphs to model network vulnerabilities, and applying Bayesian networks to perform cybersecurity risk

analysis. Sun *et al.* [19] used Bayesian networks to model the potential attack paths in the system and proposed an attack path optimization algorithm based on attacker's knowledge and attack patterns in the attack graph, thus conducting security risk assessment. In this work, the node is given a probability value to describe the probability of an attack occurring at the node and the probability value of the system is destroyed by the Bayesian network.

The above work can only deal with the simpler situation in the network system and is the static security risk assessment. Although the results of the static security risk assessment are accurate, due to the uncertainty and suddenness of the network security incidents, the evaluation results are relatively lagging and it is difficult to meet the actual needs [17]. In response to this problem, Li *et al.* [15] introduced a Bayesian attack graph model and based on this, the authors proposed a dynamic security risk assessment method. The fundamental difference between their work and our work is that they do not fully consider the environmental characteristics of the enterprise when assessing cybersecurity risks.

3 Security Vulnerability Assessment

Security Vulnerability assessment is the basis for cybersecurity risk assessment. To assess the risk of vulnerabilities associated with the environmental characteristics of an enterprise, we first introduce a set of security vulnerability assessment metrics that determined by the economic loss caused by exploits to the enterprise, and then introduce the integration of CVSS metrics, enterprise's economic loss metrics and quantitative metrics for enterprise security requirements to quantify the risk of security vulnerabilities.

3.1 Enterprise's Economic Loss Metrics

Corporate's economic loss metrics focus on the economic impact of exploits on businesses, with the goal of specifically quantifying the damage caused by cyber attacks into financial data. Before describing the metric set in detail, we first introduce several necessary conditions:

1) After the introduction of the new measurement standard, the comprehensive score of the security vulnerability should be diversified, that is, it should avoid the excessive concentration of vulnerability risk score;

2) The vulnerability risk scoring process should not be too complicated referring to the CVSS scoring principle;

3) For ease of understanding, the score should be consistent between different analysts in the company.

3.1.1 Enterprise's Economic Loss Classification

The quantitative scoring process is more objective than qualitative ratings. However, quantitative scoring does not give a relatively straightforward understanding of the risk of security vulnerability. Referring to the CVSS vulnerability risk classification principle, this paper divides the economic loss into four scales, namely low-level, intermediate-level, advanced-level and severe-level. There are two advantages to this: One is to facilitate the economic loss caused by different attack scenarios within the company; the other is to facilitate the understanding of non-technical personnel, such as business management personnel. Since it is impossible to compare the absolute value of property damage between enterprises of different scales, for example, the property loss of 100,000 US dollars may be a high-level loss for a small and medium-sized enterprise, but it may be a low-level loss for a large multinational company. Therefore, the proposed qualitative level of property loss is related to the specific financial system of a specific enterprise. The enterprise needs to define the currency interval threshold according to its own characteristics, as shown in Table 1, where the quantitative score is in decimal.

Table 1: Enterprise economic loss levels

low	$[0, C_{medium}]$	3.5
medium	$[C_{medium}, C_{high}]$	6.1
high	$[C_{high}, C_{critical}]$	7.1
serious	$[C_{critical}, \infty]$	10.0

3.1.2 Enterprise's Economic Loss Metrics

We define a set of vulnerability economic loss metrics based on the empirical work of Spagnuelo *et al.* [22]. Spagnuelo *et al.* defined economic cost units based on publicly known security incidents. This paper integrates the "potential economic loss" as shown in Figure 1. The definitions and calculation formulas for each type are described below.

Definition 1. *Revenue loss (RevL). Computer systems bring benefits to enterprises. Suppose c represents the number of customers in a business and r represents the average customer revenue for a transaction. There are two main reasons for the loss of corporate's revenue: One is that system services are not available; the other is customers' loss due to longer service response time. Suppose A indicates the availability of system services, $A = 1$ indicates that system services are available and $A = 0$ indicates that system services are unavailable. Then the loss of revenue due to the unavailability of system services is:*

$$\mathrm{Rev}L = c \times r \times (1 - A) \tag{1}$$

Definition 2. *Reputation loss (RL). The reputational damage caused by exploits is harder to measure. The usual*

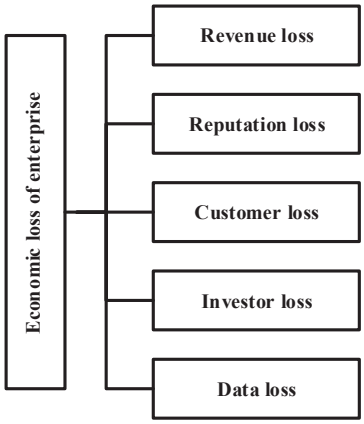


Figure 1: Economic loss of the enterprise

measure of reputational loss is by measuring the historical impact of exploits and security incidents on corporate stocks. Assuming that ise is the average historical impact of exploits on a company's stock price, then the reputation loss is calculated:

$$ise = \frac{1}{n}\sum_{t=0}^{n} P_t - P_{after} \tag{2}$$

Definition 3. *Customer loss (CL). After the enterprise's exploit event is announced, the security-sensitive customers will terminate the cooperation with the enterprise, which will lead to customer losses. The calculation formula is:*

$$CL = ssc \times arc_t \tag{3}$$

where ssc is the number of customers who are sensitive to security and arc is the average customer's revenue in each time period.

Definition 4. *Investor loss (IL). After the company's exploits are announced, security-sensitive investors will stop investing in the company. The formula for calculating investors' losses is:*

$$IL = ssi \times ai_t \tag{4}$$

where ssi is the number of security-sensitive investors and ai_t is the average investment amount of the investor in each time period.

Definition 5. *Data loss (DL). Data leakage will cause property damage to the company. The calculation formula for data loss due to data leakage is:*

$$DL = avr \times nlr \tag{5}$$

where avr is the average value of each data record and nlr is the number of lost data records. The avr value can be determined by using historical audit data within the enterprise.

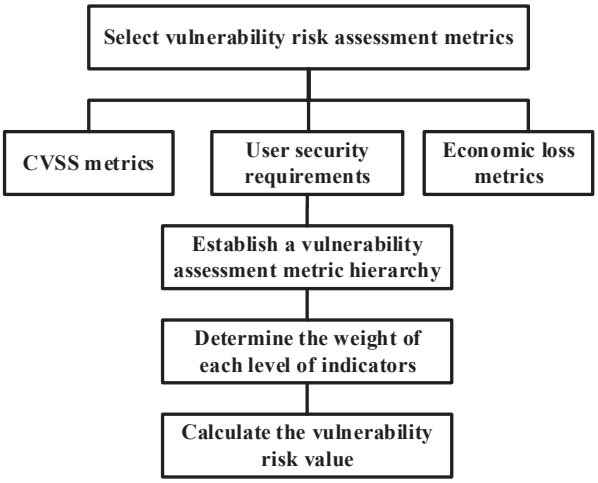


Figure 2: Risk assessment method for security vulnerabilities

3.2 Quantitative Assessment Method for Risk of Security Vulnerabilities

This paper considers the risk of vulnerabilities from three aspects: Economic losses caused by vulnerabilities, enterprise security requirements and CVSS scores of vulnerabilities for the vulnerability risk assessment based on the economic losses caused by cyber attacks to enterprises. The metrics for assessing the vulnerability risk are:

1) Corporates' economic loss metrics.

2) Enterprises' security requirements metrics.

3) CVSS basic metrics.

Because these metrics do not affect the risk assessment of vulnerabilities on average, they need to be weighted by a user-centric approach that considers the security needs of specific users and the specificities of the enterprise environment. All three types of metrics use "cost" as the sole criterion, that is, in an ideal situation, how to minimize the cost loss caused by the vulnerability. Therefore, this is a typical multi-criteria decision-making analysis (MCDA) which sorts a certain number of objects according to established standards [10]. In this article, vulnerabilities are objects that need to be sorted according to standards. The analytic hierarchy process (AHP) is one of the most widely used and accurate MCDA methods [16]. The method is divided into three levels: Target layer, criterion layer and solution layer according to the overall goal and decision-making scheme of the problem, and then the method of pairwise comparison is used to determine the importance of the decision-making scheme, so as to make a satisfactory decision. AHP can be divided into the following four steps:

1) Identify problems and establish a hierarchy;

2) Construct a judgment matrix;

3) Hierarchical single sorting and consistency test;

4) Hierarchical total ordering and combination consistency test.

According to the four steps, the established risk vulnerability assessment method is shown in Figure 2.

4 Cybersecurity Risk Assessment

In this section, we introduce the cybersecurity risk assessment method which is based on the quantitative assessment of vulnerability risk. First, we introduce the relevant definitions, then introduce the assessment method that can be combined with the characteristics of the enterprise environment for dynamic security risk assessment.

4.1 Related Definitions

Definition 6. *Atomic attack. Suppose S is a set of network attributes, A is a conditional dependency between a pair of network attributes and A is represented as a form of mapping $S \times S \to [0,1]$. Then, given $S_{pre}, S_{post} \in S$, $a : S_{pre} \to S_{post}$ is called an atomic attack if:*

1) $S_{pre} \neq S_{post}$;

2) $A(S_{pre}, S_{post}) > 0$ when $S_{pre} = 1$ and $S_{post} = 1$;

3) There does not exist $S_1, S_2, ..., S_j \in S - \{S_{pre}, S_{post}\}$ making $A(S_{pre}, S_1) > 0$, $A(S_1, S_2) > 0$, ..., $A(S_j, S_{post}) > 0$.

An atomic attack indicates that the attacker successfully reached attribute S_{post} from attribute S_{pre} with a non-zero probability. Among them, the condition 3 indicates that the attacker directly reaches the attribute S_{post} from the attribute S_{pre}, and does not pass other network attributes in the middle. In addition, an atomic attack is usually associated with an exploit which exploits an attacker from one network property to another. We use e_i for vulnerability utilization and $t(e_i)$ for the danger of exploiting the vulnerability.

Definition 7. *Bayesian attack graph (BAG). Suppose S is a set of network attributes and A the set of atomic attacks defined on S. A Bayesian attack graph is a quad of $BAG = (S, \tau, \varepsilon, P)$, where:*

1) $S = N_{internal} \cup N_{external} \cup N_{terminal}$. $N_{external}$ is a set of attributes S_i, for the set of S_i, there does not exist $a \in A | S_i = post(a)$. $N_{internal}$ is a set of attributes S_j and there does not exist $a_1, a_2 \in A | S_j = pre(a_1) \wedge post(a_2)$. $N_{terminal}$ is a set of attributes S_k and there does not exist $a \in A | S_k = pre(a)$.

2) $\tau \subseteq S \times S$. If $S_{pre} \to S_{post} \in A$, then ordered pair $(S_{pre}, S_{post}) \in \tau$. In addition, for $S_i \in S$, the set $Pa[S_i] = \{S_j \in S | (S_j, S_i) \in \tau\}$ is called the parent node set of S_i.

3) ε is a set of elements of the form $\langle S_j, d_j \rangle$. For all $\forall S_j \in N_{internal} \cup N_{terminal}$ and $d_j \in \{AND, OR\}$, d_j is AND if $S_j = 1 \Rightarrow \forall S_i \in Pa[S_j]$, $S_i = 1$. d_j is OR if $S_j = 1 \Rightarrow \exists S_i \in Pa[S_j]$, $S_i = 1$.

4) P is a set of conditional probability distributions. Each attribute $S_j \in N_{internal} \cup N_{terminal}$ has a conditional probability distribution with a value of $Pr(S_j | Pa[S_j])$.

Definition 8. *Condition probability distribution (CPD). Let $BAG = (S, \tau, \varepsilon, P)$ be a Bayesian attack graph, $S_j \in N_{internal} \cup N_{terminal}$. For $S_i \in Pa[S_j]$, e_i is an exploit related to the atomic attack $S_i \to S_j$. The conditional probability distribution of S_j is $Pr(S_j | Pa[S_j])$ and the definition is:*

if $d_j = AND$,

$$Pr(S_j | Pa[S_j]) = \begin{cases} 0, \exists S_i \in Pa[S_j] | S_i = 0 \\ t\left(\bigcap_{S_i=1} e_i\right), otherwise \end{cases} \quad (6)$$

if $d_j = OR$,

$$Pr(S_j | Pa[S_j]) = \begin{cases} 0, \forall S_i \in Pa[S_j] | S_i = 0 \\ t\left(\bigcup_{S_i=1} e_i\right), otherwise \end{cases} \quad (7)$$

When multiple exploits are involved, in order to calculate the conditional probability distribution, we proceed as follows: For the case of "AND", each exploit is an independent event. The probability of destroying a target node depends on the probability of successfully exploiting a single exploit. Therefore, the law of independence of events is:

$$t\left(\bigcap_{S_i=1} e_i\right) = \prod_{S_i=1} t(e_i). \quad (8)$$

In the case of "OR", this relationship is actually a Noisy-OR relationship. There is:

$$t\left(\bigcup_{S_i=1} e_i\right) = 1 - \prod_{S_i=1} [1 - t(e_i)]. \quad (9)$$

4.2 Cybersecurity Risk Assessment Method

Cybersecurity risk assessment is the basis for network security risk management. Currently, cybersecurity risk assessment techniques can be divided into two categories: Static security risk assessment and dynamic security risk assessment.

The static security risk assessment is to evaluate the security risks of the network in a short period of time or at a certain point in time. Although the assessment results are accurate, they are relatively lagging, so it is difficult to meet the actual needs. Dynamic security risk assessment studies the evolution trend of network security risks

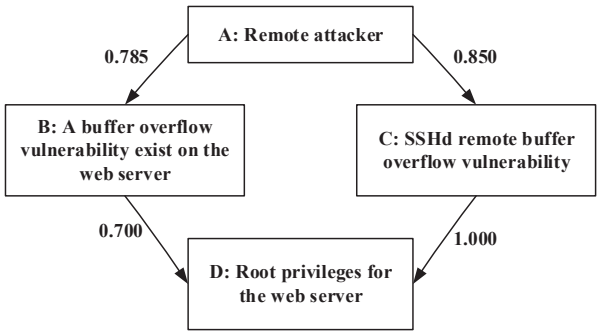


Figure 3: Risk assessment method for security vulnerabilities

and evaluates the network security in a period of time in combination with the changes of network environment, so as to grasp the changes of network security risks with the changes of network environment factors. The dynamic security risk assessment method we use is described below.

During the life cycle of a network system, the probability of occurrence of each network state changes. Emerging cybersecurity events can affect the likelihood of an attack. This paper evaluates network security risks from these emerging cybersecurity incidents by using the Bayesian attack graph model to calculate posterior probabilities.

Suppose $S = \{S_1, S_2, ..., S_n\}$ is a set of attributes in a Bayesian attack graph and $E = \left\{S_1^{'}, S_2^{'}, ..., S_m^{'}\right\} \subset S$ is a subset of S, the attributes in this set represent the attack events that have occurred. These attributes are called "evidence", i.e. for all $S_i^{'} \in E$, there is $S_i^{'} = 1$. Existing S_j needs to determine the posterior probability of S_j. According to Bayes' theorem, there is:

$$\Pr\left(S_j|E\right) = \frac{\Pr\left(E|S_j\right) \times \Pr\left(S_j\right)}{\Pr\left(E\right)} \tag{10}$$

where $\Pr\left(E|S_j\right)$ is the conditional probability that $\left\{S_1^{'}, S_2^{'}, ..., S_m^{'}\right\}$ are combined in the state given S. $Pr(E)$ and $Pr(S_j)$ are priori unconditional probability values for the corresponding attributes. The evidence in E is independent of each other, so we have $\Pr\left(E|S_j\right) = \prod_i \Pr\left(S_i^{'}|S_j\right)$ and $\Pr\left(E\right) = \prod_i \Pr\left(S_i^{'}\right)$.

5 Case Study

5.1 Case Analysis

This paper takes a small bayesian attack graph shown in Figure 3. as an example to illustrate the calculation process of the network security risk assessment method in detail. In Figure 3. node A represents "remote attacker", node B represents "a buffer overflow vulnerability exists on the web server (CVE-2019-9933)" and node C represents "SSHd remote buffer overflow vulnerability". Node D stands for "root privileges for the web server." The edges in Figure 3. indicate the corresponding exploits.

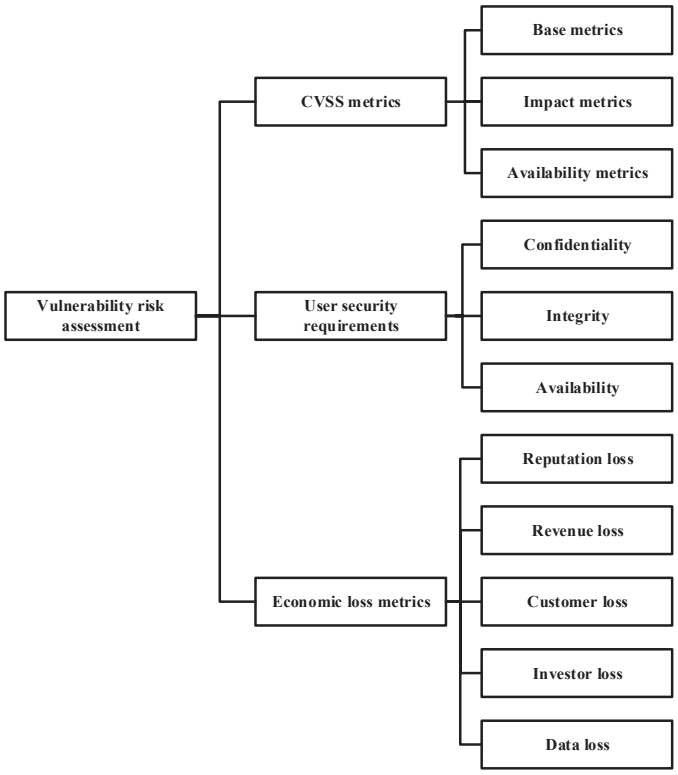


Figure 4: Hierarchical structure of vulnerability severity assessment

For example, the edge between node A and node B indicates that "the attacker exploited the buffer overflow vulnerability to launch an attack." The value next to each edge is the result of dividing the vulnerability risk quantified value by 10, in order to make the score between 0 and 1. The attacker's goal is to gain root access to the Web server for damage. It is assumed that in this case, the availability of the Web server is high, and the security risk assessment of the network structure is performed for this feature. The specific calculation process is as follows.

Step 1. Establish a hierarchy of vulnerability risk assessments.

The hierarchy of the risk assessment for vulnerability CVE-2019-9933 is shown in Figure 4.

Step 2. Construct a judgment matrix.

In this case, the availability of Web servers is high, so the CVSS metrics, user security requirements and economic loss metrics are constructed in a 1:3:1 ratio. The importance of the criteria layer for the target layer is $G = \begin{bmatrix} 1 & 1/3 & 1 \\ 3 & 1 & 3 \\ 1 & 1/3 & 1 \end{bmatrix}$.

Similarly, the importance matrices of the solution layer for the criterion layer are $C_1 = \begin{bmatrix} 1 & 4 & 4 \\ 1/4 & 1 & 1 \\ 1/4 & 1 & 1 \end{bmatrix}$, $C_2 = \begin{bmatrix} 1 & 1 & 1/3 \\ 1 & 1 & 1/3 \\ 3 & 3 & 1 \end{bmatrix}$, and $C_3 =$

$\begin{bmatrix} 1 & 2 & 3 & 5 & 6 \\ 1/2 & 1 & 3 & 4 & 5 \\ 1/3 & 1/3 & 1 & 3 & 5 \\ 1/5 & 1/4 & 1/3 & 1 & 2 \\ 1/6 & 1/5 & 1/5 & 1/2 & 1 \end{bmatrix}$. Considering that the data loss accounts for the largest proportion of economic losses, the loss of revenue, reputation loss, customer loss, investor loss and data loss construct the matrix C_3 in a ratio of $1:2:3:5:6$.

Step 3. Hierarchical single sorting and consistency check. Taking the judgment matrix G as an example, we use Matlab to calculate the maximum eigenvalue, the corresponding eigenvector, the consistency index and the random consistency ratio of the matrix G are $\lambda_{\max} = 4$, $W = (0.15, 0.45, 0.1)^T$, $C = \frac{\lambda_{\max}-n}{n-1} = \frac{4-4}{4-1} = 0$ and $C_R = 0$ respectively. Therefore, the CVSS base metrics, user security requirements and economic loss metrics can be considered to have weights of vulnerability risk assessments of 0.1, 0.45, and 0.35 respectively.

Step 4. Hierarchical total ordering and its combination consistency test. Hierarchical total sorting combination consistency check $C = 0$, $C_R = 0 < 0.1$.

Step 5. Calculate the vulnerability value of vulnerability risk. According to the expert's scoring sample matrix of vulnerability CVE-2019-9933, the risk of the vulnerability is quantified as:

$$P_v = W \times S = 8.6428 \tag{11}$$

The risk of other vulnerabilities in Figure 3. can be quantified using the same method.

Step 6. Calculation of node risk value.

Suppose the network administrator detects a network attack on node D, that is, the attacker gains root access to the web server. The posterior probability of node B is calculated as follows:

$$\Pr(B|D) = \frac{\Pr(D|B)\Pr(B)}{\Pr(D)} \tag{12}$$

where,

$$\Pr(D|B) = \sum_{C\in\{T,F\}} [\Pr(D|C, B = T)\Pr(C)] \tag{13}$$

The posterior probability of the Node B is 0.6830. It is worth noting that the node's unconditional probability is 0.4810 without considering the web server being attacked. After considering the attack event occurring on node D, the posterior probability of node B becomes 0.6830. There is a significant improvement over the previous one. By taking into account the environmental information of the system, it is possible to make a more accurate and effective assessment of the security of the network.

5.2 Effect Evaluation

This section uses the network topology shown in Figure 5. as the evaluated network for simulation experiment. We suppose a small and medium-sized enterprise X specializes in providing online electronic trading services to users. The enterprise's network topology is shown in Figure 5. The network consists of three sub-networks, namely the external service area, the internal management area and the internal user area. The three areas are divided by a firewall, and the entire network is connected to the Internet through a gateway. Among them, the external service area mainly includes a network server and a mail server. The two servers provide network services and mail services to external and internal users respectively. The internal management area includes a file transfer server, two database servers and two clients. The file transfer server mainly provides web-related file storage and management services for the web server and the two clients can operate the file transfer server through SSH links. Potential attackers on the network come from external attackers accessing the Internet. We use Nessus as a vulnerability scanning tool to obtain vulnerability information on each host/server in the network as shown in Table 2.

Taking into account the characteristics of enterprise X, we simulate two application scenarios for experimental analysis.

Scenario 1. In this network, the web server is just a common web server for publishing common sense and introductory, without storing important and valuable data and information. In this case, the enterprise has high demand for the availability of the network server.

Scenario 2. In this network, the network server bears the main service of the network, and the collapse of the network server will have a greater impact on the enterprise. In this case, the enterprise has higher requirements for the availability and confidentiality of the network server.

We use the proposed cybersecurity risk assessment method (NSRAEE) to calculate the risk quantified values of the servers in these two scenarios and use the method of [15] to calculate the risk quantified values of the servers in the two scenarios. The results are shown in Table 3.

Table 3 shows the calculation results of the server risk quantized values in the two scenarios of this method and the calculation results of the reference method. The reference method has the same result in both scenarios, so only a set of results is shown. It can be seen from Table 3 that the method of [15] does not consider the security requirements of the enterprise network and the risk quantified values of the respective servers calculated in the two scenarios are the same. With this method, the server's risk quantification value will vary depending on the enterprise environment. For example, in scenario 2, the network server assumes the main service of the network and is an important business asset of the enterprise.

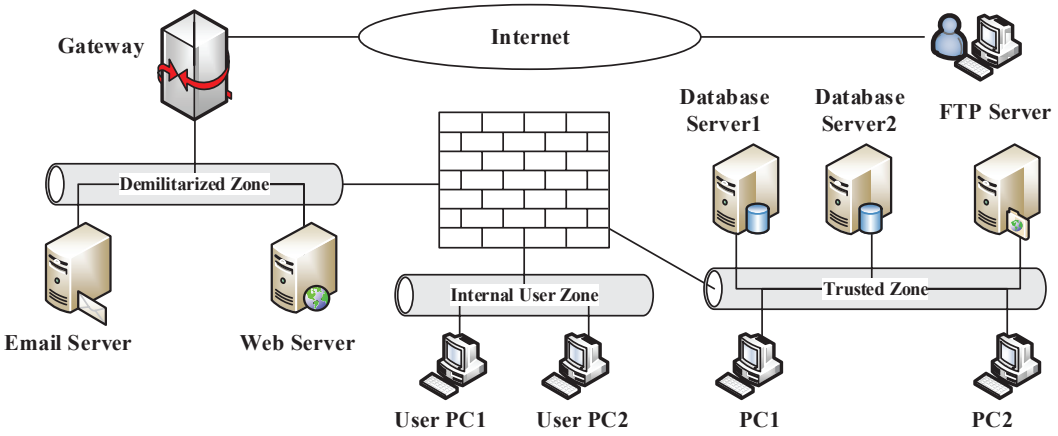


Figure 5: Network topology

Table 2: Vulnerability information

host	CVE number	Attack type
network server	CVE-2019-8952	DDoS
mail server	CVE-2019-12497	remote attack
	CVE-2019-10735	information leakage
ftp server	CVE-2019-10967	privilege escalation
	CVE-2018-7240	remote attack
	CVE-2019-11380	remote attack
database server 1	CVE-2019-5632	remote attack
database server 2	CVE-2019-7667	remote attack
gateway	CVE-2019-8319	information leakage

Table 3: Risk quantification values of servers

Host	Reference method	NSRAEE: Scenario 1	NSRAEE: Scenario 2
network server	0.6127	0.3891	0.7628
mail server	0.6854	0.5588	0.7359
ftp server	0.6987	0.4847	0.7983
database server 1	0.6218	0.3456	0.8742
database server 2	0.6142	0.5754	0.6877
gateway	0.5874	0.6683	0.7531

If it collapses or is invaded, it will have a greater impact on the enterprise. Therefore, the network server obtained using the calculation method of this paper has a larger risk quantized value in scenario 2 than in scenario 1.

In summary, the proposed network security risk assessment method considers the security requirements of the enterprise network environment and covers the impact of environmental threat information on the node risk, making the method more suitable for the network or information system being evaluated. The actual situation of the possibility of an attack is more objective and accurate.

6 Conclusion

In the process of cybersecurity management, cybersecurity risk assessment is the premise and foundation of network security management. In order to develop a reliable cybersecurity risk assessment method that is consistent with the actual situation of the enterprise, it is necessary to fully consider the environmental characteristics of a specific enterprise. In response to this problem, this paper proposes a method to assess the security risks of enterprise network systems based on the characteristics of enterprise environment. First, we assess the risk of security breaches based on enterprise security needs, the economic losses caused by the attack and the CVSS base metric. Then, we use the Bayesian attack graph model combined

with the environmental changes of the enterprise network system for dynamic security risk assessment. Finally, the specific calculation process is illustrated by the case study, and the simulation experiment proves that compared with the existing methods, the quantitative evaluation method proposed in this paper is more suitable for the safety risk status of the evaluated enterprise. The evaluation result is more objective and accurate.

Future research work will further consider more possible metrics for corporate economic losses. In addition, how to simplify the scale of the attack graph is also the focus and difficulty of the research.

7 Acknowledgments

This work is supported by The Natural Science Foundation of the Jiangsu Higher Education Institutions of China (No.19KJD520002), and Jiangsu Province Higher Education College Students Innovation and Entrepreneurship Training Program (No. 202012078001Y).

References

[1] Allianz Risk Barometer 2020 - Top global business risks, 2021. (`https://www.agcs.allianz.com/news-and-insights/reports/allianz-risk-barometer.html`)

[2] S. Atapattu, N. Ross, Y. Jing, *et al.*, "Physical-layer security in full-duplex multi-hop multi-user wireless network with relay selection," *IEEE Transactions on Wireless Communications*, vol. 18, pp. 1216–1232, 2019.

[3] Y. H. Chen, Y. Z. Xie, X. Y. Ge, *et al.*, "Vulnerability assessment of equipment excited by disturbances based on support vector machine and gaussian process regression," *IEEE Transactions on Electromagnetic Compatibility*, no. 99, pp. 1-8, 2020.

[4] W. U. Chensi, T. Wen, Y. Zhang, "A revised CVSS-based system to improve the dispersion of vulnerability risk scores," *Information Sciences*, vol. 62, no. 03, pp. 193-195, 2019.

[5] Common Vulnerability Scoring System v3.1. (`https://www.first.org/cvss/v3-1/`)

[6] B. Cruz, S. Gomez-Meire, D. Ruano-Ordas, *et al.*, "A practical approach to protect IoT devices against attacks and compile security incident datasets," *Scientific Programming*, vol. 2019, no. 4, pp. 1-11, 2019.

[7] S. Ding, Z. Zhang, J. Xie, "Network security defense model based on firewall and IPS," *Journal of Intelligent and Fuzzy Systems*, no. 12, pp. 1-9, 2020.

[8] M. Gao, J. Zhang, J. Yu, *et al.*, "Recommender systems based on generative adversarial networks: A problem-driven perspective," *Information Sciences*, vol. 546, pp. 1166-1185, 2020.

[9] N. M. Karie, V. R. Kebande, H. S. Venter, "Diverging deep learning cognitive computing techniques into cyber forensics," *Forensic Science International: Synergy*, vol. 1, pp. 61-67, 2019.

[10] P. Lin, Y. Chen, "Network security situation assessment based on text SimHash in big data environment," *International Journal of Network Security*, vol. 21, no. 4, pp. 699-708, 2019.

[11] S. Mahdavifar, A. A. Ghorbani, "DeNNeS: Deep embedded neural network expert system for detecting cyber attacks," *Neural Computing and Applications*, vol. 32, no. 6, 2020.

[12] A. Olagunju, "Where did I leave my keys?: Lessons from the Juniper dual EC incident," *Computing Reviews*, vol. 60, no. 4, pp. 168-169, 2019.

[13] S. Rosli, R. S. Abdullah, W. Mohamed, "Ransomware behavior attack construction via graph theory approach," *International Journal of Advanced Computer Science and Applications*, vol. 11, no. 2, pp. 10, 2020.

[14] M. Y. Ruan, H. D. Chiang, "On the accuracy of the online static security assessment under Different models: Assessment and basis," *IEEE Transactions on Power Systems*, vol. 99, pp. 1-8, 2019.

[15] S. S. Sathya, K. Umadevi, "An optimized distributed secure routing protocol using dynamic rate aware classified key for improving network security in wireless sensor network," *Journal of Ambient Intelligence and Humanized Computing*, pp. 1-7, 2020. DOI:10.1007/s12652-020-02392-2.

[16] K. Senthilkumar, R. Ramadoss, "Optimized scheduling of multicore ECU architecture with bio-security CAN network using AUTOSAR," *Future Generation Computer Systems*, vol. 98, pp. 1-11, 2019.

[17] C. Shen, Y. Chen, X. Guan, *et al.*, "Pattern-growth based mining mouse-interaction behavior for an active user authentication system," *IEEE Transactions on Dependable and Secure Computing*, vol. 17, no. 2, pp. 335-349, 2020.

[18] N. Sun, J. Zhang, P. Rimba, *et al.*, "Data-driven cybersecurity incident prediction: A survey," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 2, pp. 1744-1772, 2019.

[19] D. Wang, T. Muller, J. Zhang, *et al.*, "Information theoretical analysis of unfair rating attacks under subjectivity," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 816-828, 2020.

[20] Y. Xiao, Z. J. Fan, A. Nayak, *et al.*, "Discovery method for distributed denial-of-service attack behavior in SDNs using a feature-pattern graph model," *Frontiers of Information Technology & Electronic Engineering*, vol. 20, no. 9, pp. 1195-1208, 2019.

[21] B. Yang, W. Bao, Y. Chen, "Time series prediction based on complex-valued S-system model," *Complexity*, vol. 24, pp. 1-13, 2020.

[22] J. Zhang, Y. Chen, Y. Zhai, "Zero-shot classification based on word vector enhancement and distance metric learning," *IEEE Access*, vol. 99, pp. 1-1, 2020.

[23] L. Zhu, "Safety detection algorithm in sensor network based on ant colony optimization with improved multiple clustering algorithms," *Safety Science*, vol. 118, pp. 96-102, 2019.

Biography

Yunxue Yang was born in 1986. She received the B.S. in computer science from Qufu Normal University, China, in 2007 and M.S. in computer and information science from Nanjing University of Information Science & Technology, China, in 2011. She is currently a lecturer of the Department of Computer Science and Technology, Silicon Lake College, Kunshan, China. Her current research interests include network security, cryptography and information security.

Zhenqi Yang received his B.S. in Fundamental Mathematics from Qufu Normal University, China, in 1983 and M.S. in Applied Mathematics from Chinese Academy of Sciences, China, in 1988. He is current a professor at the department of Internet of Things, Binjiang College of Nanjing University of Information Science, China. His current research interests include information security, cryptography and mobile communications.

Qin Yang received her B.S. in Computer and Information Sciences from Wuhan University, Wuhan, China, in 2006 and M.S. in industrial engineering from Southeast University, China, in 2014. She is current an associate professor at the department of Computer Science and Technology, Silicon Lake College, Kunshan, China. Her current research interests include The Internet of Things and information security.

Guohua Ji received his B.S. in Computer and Information Sciences from Shanghai Normal University, Shanghai, China, in 2003. He is current an associate professor at the department of Computer Science and Technology, Silicon Lake College, Kunshan, China. His current research interests include information security and software security.

Shengjun Xue received his Ph.D. degree in computer science and information engineering at the Zhejiang University, China. Later on, he worked at Indiana University-Purdue University Indianapolis as a post-doctoral fellow. Then, he worked at Wuhan University of Technology, China, in 2000. He now is a distinguished professor at the department of Computer Science and Technology, Silicon Lake College, Kunshan, China. His current research interests include big data and cloud computing.

A CKKS-based Privacy Preserving Extreme Learning Machine

Kunhong Li and Ruwei Huang
(Corresponding author: Ruwei Huang)

School of Computer and Electronic Information, Guangxi University
Nanning 530004, China
Email: ruweih@126.com

(Received Feb. 5, 2021; Revised and Accepted Aug. 6, 2021; First Online Dec. 18, 2021)

Abstract

As an algorithm of machine learning, extreme learning machine has good classification and prediction performance in shallow models. Still, there is a security problem of user data leakage in the cloud environment. We propose a new privacy-preserving extreme learning machine based on the CKKS homomorphic encryption (HE) scheme. As a result, users can send encrypted data to service providers for analysis and prediction without revealing their data privacy. We also designed a Single Instruction Multiple Data (SIMD) method adapted to the extreme learning machine, which reduces the number of homomorphic operations and user's waiting time. Through experiments, this paper verifies the superiority of CKKS-ELM in efficiency and accuracy.

Keywords: CKKS; Cloud Service; Extreme Learning Machine; SIMD

1 Introduction

With the development of artificial intelligence in recent years, Machine Learning as a Service (MLaaS) has been proposed, which provides Machine Learning tools as a part of cloud computing services for users to use, including image processing, speech recognition, and data analysis. However, the service provider is not necessarily honest, and it may steal the user's information. For example, the hospital needs to send the images to the cloud server for predictive analysis of some patients' symptoms. However, the data is very sensitive, and sending the images to a third party may have privacy disclosure problems.

Extreme learning machine(ELM) is a simple and efficient single hidden layer feedforward neural network learning algorithm proposed by Huang *et al.* [13], Which can quickly learn the characteristics of data sets, and has good performance in shallow classification model. It has been widely used in disease diagnosis, Traffic Sign Recognition, and image quality evaluation.

To protect data privacy, many researchers have studied the ELM for privacy protection. Samet *et al.* [22] Proposed a Privacy Protection protocol for back-propagation and ELM, based on the multi-party Secure Computing protocol they proposed, data is divided into several parts vertically or horizontally for multiple servers to process, but there are some problems such as low efficiency and large communication cost. Masato *et al.* [10,11] and proposed an ELM-based privacy preserving protocol for implementing aware agents, which protects the user's privacy. Ferhat *et al.* [5] proposed a privacy-preserving extreme learning machine based on Paillier homomorphic encryption called CPP-ELM, the training data is vertically divided into several parts, each piece of data is processed, encrypted, and transmitted to the server for integration. Yoshinori *et al.* [15] proposed a PP-ELM outsourcing scheme based on Paillier homomorphic encryption for training process of regularized ELM. Lin *et al.* [16] focuses on outsourcing ELM in cloud, and presents an optimization mechanism to improve training speed of ELM. Yang [14] proposed a privacy protection extreme learning machines based on fully homomorphic encryption scheme and carried out ciphertext image comparison experiments on medical image classification data sets.

These papers [5, 10, 11, 14–16, 22] are based on Paillier homomorphic encryption and expose the training model to the user, who can always use the model in the local environment after obtaining the model. Wang *et al.* [24] Proposed HOMO-ELM based on fully homomorphic encryption BGV scheme, which adopts HE-friendly activation function and prevents the leakage of training model. However, the amount of homomorphic operations in a single prediction is too many, which leads to a long waiting time for users and can only ensure the safety of single users.

1) We propose a privacy model scenario with high confidentiality, which is more satisfied with actual commercial application scenarios;

2) We combine the ELM with the CKKS homomorphic encryption scheme that is more suitable for floating-

point calculations and propose a better homomorphic friendly activation function to improve the classification performance;

3) According to the ELM's structure, we propose a suitable SIMD method to reduce latency during prediction.

The rest of the paper is organized as follows. In Section 2, we introduce the ELM and the CKKS fully homomorphic encryption. In Section 3, a CKKS-based ELM is proposed. In Section 4, we present experimental results and performance comparison. Finally, our conclusions are presented in Section 5.

2 Preliminaries

2.1 Extreme Learning Machine

Extreme learning machine is a single hidden layer neural network. It is composed of an input layer, a hidden layer and an output layer. The neurons of the input layer and the hidden layer, and the hidden layer and the output layer are fully connected. Let n be the neurons number of input layer, corresponding to n input features. Let l be the neurons number of the hidden layer, and let m be the neurons number of the output layer, corresponding to m classification results. Let $g(x)$ be the activation function of the hidden layer neuron, for the j-th training input data $x_j = [x_{1j}, x_{2j}, \cdots, x_{nj}]^T$, the output defines as follow:

$$T = \begin{bmatrix} t_{11} & \cdots & t_{1Q} \\ \vdots & \ddots & \vdots \\ t_{m1} & \cdots & t_{mQ} \end{bmatrix}, \tag{1}$$

$$t = \begin{bmatrix} t_{1j} \\ t_{2j} \\ \vdots \\ t_{mj} \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^{l} \beta_{i1} * g(w_i x_j + b_i) \\ \sum_{i=1}^{l} \beta_{i2} * g(w_i x_j + b_i) \\ \vdots \\ \sum_{i=1}^{l} \beta_{im} * g(w_i x_j + b_i) \end{bmatrix} \quad (j = 1, 2, \cdots, Q)$$

Where $w_i = [w_{i1}, w_{i2}, \cdots, w_{in}]$, represents the connection weight between the i-th neuron in the input layer and the j-th neuron in the hidden layer. $b_i = [b_1, b_2, \cdots, b_l]$, represents the bias value of the hidden layer neuron. β_{jk} represents the connection weight between the j-th neuron in the hidden layer and the k-th neuron in the output layer.

The Equation (1) can also be simplified as Equation (2):

$$H\beta = T^T \tag{2}$$

Because both w and b are randomly generated weights, and the target label matrix T is also known, the hidden layer output matrix β is uniquely determined. Training a single hidden layer neural network can be transformed into solving Equations (3):

$$\beta = H^+ T^T \tag{3}$$

where H^+ is Moore–Penrose generalized inverse of matrix H.

Because the ELM mainly reduces the error rate during training, it may cause the model to overfit. To solve this problem, Huang *et al.* [12] later introduced the regularization parameters C. Finally, the weight β can be expressed as Equations (4):

$$\beta = (\frac{I}{C} + H^T H)^{(-1)} H^T T. \tag{4}$$

Where I is the identity matrix.

2.2 Homomorphic Encryption

The concept of homomorphic encryption was proposed [21] in 1978, and homomorphic operations can be performed between homomorphically encrypted ciphertexts. For example, two plaintexts a and b are homomorphically encrypted to c and d, and the encryption function $Dec(Enc(a) \odot Enc(b)) = Dec(c \odot d) = a \oplus b$ is satisfied, where Enc is the encryption operation and Dec is the decryption operation, $\odot$ and $\oplus$ correspond to the operations on the plaintext and ciphertext domains, respectively. The early homomorphic encryption cryptosystem only supports addition or multiplication. Until 2009, Gentry [9] proposed the first fully homomorphic encryption scheme based on the ideal lattice. Homomorphic encryption is widely used in cloud environment for image comparison, face recognition, disease diagnosis, and various kind of data analyst [5, 10, 11, 14, 15, 17, 18, 22, 24]. But Cao *et al.* [4] stressed that some typical schemes are not suitable for cloud computing scenarios because the lack efficiency and do not naturally support decimal operations.

In 2017, Cheon *et al.* [7] proposed an approximate homomorphic encryption scheme called CKKS17. This scheme has greatly improved the computational efficiency of the floating-point number, which makes the use of homomorphic encryption in machine learning achieve ideal efficiency. In 2018, Cheon *et al.* [6] used the residual system as an optimization method based on the CKKS17 scheme, which improved the efficiency. Their implementation showed speed-ups 17.3, 6.4, and 8.3 times for decryption, constant multiplication, and homomorphic multiplication. This scheme is still one of the most efficient all homomorphic encryption schemes. Users can encrypt sensitive information and then send it to the service provider for analysis and processing. Finally, the user decrypts the ciphertext with his private key to obtain the analysis results.

2.3 The CKKS Leveled Fully Homomorphic Encryption Scheme

CKKS scheme is an approximate homomorphic encryption scheme. This scheme can maintain a certain precision in homomorphic operation. For example, $1.234 \times 2.222 = 2.741948 \approx 2.742$. Although the precision is decreased, CKKS is faster in operation speed, and this property is very consistent with the characteristics of machine learning. As long as the relative value is correct, the prediction results are correct. Since the module length of CKKS greatly affects the computational efficiency, Cheon *et al.* later proposed the residue number system (RNS) variant of CKKS scheme.

CKKS scheme is based on learning with errors(LWE) problem. Let N be a power of 2 integer and $R = Z[X]/(X^N+1)$ be the ring of polynomials modulo X^N+1. For a fixed base q and bit precision η, select the coprime chain of moduli $\{q_0, \cdots, q_L\}$such that $q/q_l \in (1-2^{(-\eta)}, 1+2^{(-\eta)})$. The level represents the modulus of the current ciphertext. In the leveled fully homomorphic encryption, the multiplication between ciphertexts will consume a level. When the ciphertext is in level 0, the ciphertext cannot do multiplication operation again. τ denotes a variant of the canonical embedding. The following is a brief description of the main operation part of the scheme. Readers can reference specific details and noise analysis in [6].

Setup$(q, L, \eta; 1^{(\lambda)})$**:** Given a base q, maximum computation levels L, bit precision η and security parameter λ, return power-of-two integer N, secret key distribution χ_{key}, encryption key distribution χ_{enc}, error distribution χ_{err} and RNS basis D.

KSGEN (s_1, s_2)**:** Given a ciphertext$(s_1, s_2) \in R$, sample $(a^{'(0)}, \cdots, a^{'(k+L)}) \longleftarrow U(\prod_{i=0}^{k-1} R_{p_i} \times \prod_{j=0}^{L} R_{q_j})$ and $e^{'} \longleftarrow \chi_{err}$. Return $swk = (swk^{(0)} = (b^{'(0)}, a^{'(0)}), \cdots, swk^{(k+L)} = (b^{'(k+L)}, a^{'(k+L)})) \in U(\prod_{i=0}^{k-1} R_{p_i}^2 \times \prod_{j=0}^{L} R_{q_i}^2)$, where $b^{'(i)} \longleftarrow -a^{'(i)} \cdot s_2 + e^{'} (\mathrm{mod}\ p_i), 0 \le i < k$, $b^{'(k+i)} \longleftarrow -a^{'(k+j)} \cdot s_2 + [P]_{q_j} \cdot s_1 + e^{'} (\mathrm{mod}\ q_j), 0 \le j \le L$.

*KeyGen***:** Sample $s \longleftarrow \chi_{key}$, $(a^{(0)}, \cdots, a^{(L)}) \longleftarrow U(\prod_{i=0}^{L} R_{q_j})$ and $e \longleftarrow \chi_{err}$, return private key $sk = (1, s)$, switching key $swk \longleftarrow KSGEN(s^2, s)$, public key $pk \longleftarrow (pk^{(j)} = (b^{(j)}, a^{(j)}) \in R_{q_j}^2)_{0 \le j \le L}$.

Enc(m, pk, p)**:** Given a plaintext message $m \in \mathbb{C}^{\frac{N}{2}}$ and precision ρ, public key pk. Sample $v \longleftarrow \chi_{enc}$ and $e_0, e_1 \longleftarrow \chi_{err}$. Return $ct \in \prod_{j=0}^{L} R_{q_j}^2$, where $ct^{(j)} \longleftarrow v \cdot pk^{(j)} + (\tau^{-1}(2^p \cdot m + e_0, e_1) (\mathrm{mod}\ q_j),\ 0 \le j \le L$.

Dec(ct, sk, p)**:** Given a ciphertext message $ct \in \prod_{j=0}^{l} R_{q_j}^2$ and secret key sk, return $\tau(2^{-p} \cdot < ct^{(0}, sk >) (\mathrm{mod}\ q_0)$.

Add$(ct, ct^{'})$**:** Given a ciphertext message $ct \in \prod_{j=0}^{l} R_{q_j}^2$ and $ct^{'} \in \prod_{j=0}^{l} R_{q_j}^2$, return a ciphertext message $ct_{add} \in \prod_{j=0}^{l} R_{q_j}^2$, where $ct_{add} \longleftarrow ct + ct^{'} (\mathrm{mod}\ q_j), 0 \le j \le l$.

Mult$(ct, ct^{'})$**:** Given a ciphertext message $ct \in \prod_{j=0}^{l} R_{q_j}^2$ and $ct^{'} \in \prod_{j=0}^{l} R_{q_j}^2$, after the relinearization in RNS form, finally return a ciphertext $ct_{mult}^{(j)} \in \prod_{j=0}^{l} R_{q_j}^2$.

CMult(ct, v, p)**:** Given a ciphertext message $ct \in \prod_{j=0}^{l} R_{q_j}^2$ and vector v, $ct^{'} = \tau^{(-1)}(2^p \cdot v) \in R_{q_j}^2$, where $ct_{CMult}^{(j)} \longleftarrow ct^{(j)} \cdot ct^{'(j)} (\mathrm{mod}\ \ q_j), 0 \le j \le l$.

Rescale(ct)**:** Given a level-l ciphertext $ct = (ct^{(j)} = (ct_0^{(j)}, ct_1^{(j)}))_{0 \le j \le l}$, return $ct^{'} \in \prod_{j=0}^{l-1} R_{q_j}^2$, where $ct_i^{'(j)} = q_l^{-1} \cdot (c_i^{(j)} - c_i^{(l)}) (\mathrm{mod}\ q_j), i = \{0, 1\}$ *and* $0 \le j < l$.

3 CKKS-ELM

3.1 Model Training

The PP-ELM scheme proposed a safe model training protocol. Without exposing the data sets of all parties, the data of all parties can be aggregated, and these data can be used for training on the ELM, and finally, a complete training model can be obtained. But the premise of the original scheme is to assume that the data analyst is honest and will not steal intermediate data from the outsourced server. Let f be the dimension of data, l be the number of hidden nodes, k be the number of classes, N be the amount of record. The data provider will not leak data information when the following conditions are satisfied:

$$\begin{cases} Nf > \frac{l^2+l}{2} \\ N(f+k) > \frac{l^2+l}{2} + l \cdot k \end{cases}$$

Because the data analyst has n equations with at least $n+1$ unknown values. That is to say, each data provider needs to provide a data amount N greater than $max(\frac{l^2+l}{2f}, l)$ and carry out sum operation locally and then upload it to the cloud server, making it impossible to obtain any data even if the plaintext is decrypted. For details of sum operation, please refer to the paper [15]. Notably, the activation function used during training needs to be HE-friendly.

3.2 Tanhre Polynomial Approximation

In neural networks, nonlinear functions have stronger expressive ability than linear functions, thereby improving neural networks' classification performance. Common activation functions include Sigmoid, Tanh, and ReLu. The sigmoid function is widely used in ELM, but it compresses the data in the interval $(0, 1)$. In contrast, Tanh compresses the data in $(-1, 1)$ On the interval. Relu directly removed the negative part of the data and only kept

the positive one. Accordingly, Matthew *et al.* [19]combined the advantages of Tanh and Relu and designed a new activation function TanhRe, which has the following two characteristics: (1) Replace the negative part of Relu with Tanh, thereby improving network classification performance;(2) Negative values have boundary limits, while positive partdo not. TanhRe is defined as follows:

$$f(x) = \begin{cases} Tanh(x) & x < 0 \\ x & x \geq 0 \end{cases}$$

Dian *et al.* [20] used ELM to classify active compounds in Simplified Molecular Input Line Entry System (SMILES) and compared several activation functions. TanhRe was found to be one of the best activation functions. Therefore, TanhRe is used in this scheme to compare with previous HOMO-ELM schemes. Because the scheme based on fully homomorphic encryption only supports polynomial operation, and the common activation functions are nonlinear functions, the traditional activation function cannot be directly used under homomorphic operation, and the nonlinear activation function can only be expressed by polynomial approximation. The common approximation methods include Taylor expansion, Lagrange interpolation and least squares approximation. In this paper, the least squares approximation is adopted to ensure that the error between the approximation function and the original function in the specified interval is small. Suppose the fitting polynomial $\sum_{i=0}^{n} a_i \cdot x^i$, and the highest degree of polynomial is k. For a given n points (x_i, y_i), the problem is reduced to solving the following system of Equation (5):

$$\begin{bmatrix} n & \sum_{i=1}^{n} x_i & \cdots & \sum_{i=1}^{n} x_i^k \\ \sum_{i=1}^{n} x_i & \sum_{i=1}^{n} x_i^2 & \cdots & \sum_{i=1}^{n} x_i^{k+1} \\ \vdots & \vdots & \ddots & \vdots \\ \sum_{i=1}^{n} x_i^k & \sum_{i=1}^{n} x_i^{k+1} & \cdots & \sum_{i=1}^{n} x_i^{2k} \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_k \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^{n} y_i \\ \sum_{i=1}^{n} x_i y_i \\ \vdots \\ \sum_{i=1}^{n} x_i^k y_i \end{bmatrix} \tag{5}$$

The polynomial approximation of TanhRe is obtained by the least square method. $tr_3\ (x)$ and $tr_7\ (x)$ denote the approximate polynomial of TanhRe in the $(-5, 5)$ interval, respectively. Table 1 shows the results of TanhRe approximation. Figure 1shows a comparison between TanhRe and its polynomial approximation.

3.3 Data Encoding

To pack data into the same ciphertext for parallel calculation, the data needs to be re-encoded. w denotes the weight matrix from the input layer to the hidden layer and β denotes the weight matrix from the hidden layer to the output layer. In the homomorphic encryption scheme, let $N = 2 * len * wid, C = N/2$, where len and wid are both powers of 2. Therefore, w and β are divided into (l/C) submatrices z as follows, where l denotes the weight

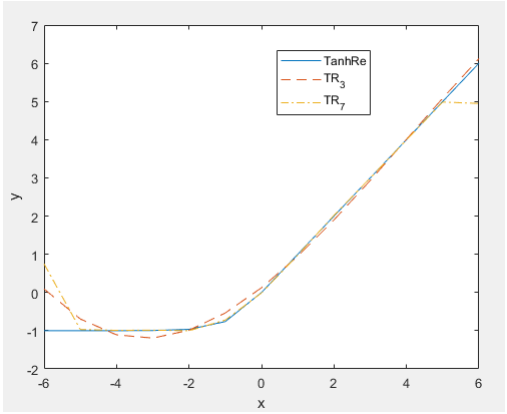


Figure 1: Comparison of Tanhre and Tanhre polynomial approximation

amount of w or β.

$$z = \begin{bmatrix} z_1 & z_2 & \cdots & z_{wid} \\ z_{wid+1} & z_{wid+2} & \cdots & z_{2*wid} \\ \vdots & \vdots & \ddots & \vdots \\ z_{(len-1)*wid+1} & z_{(len-1)*wid+2} & \cdots & z_{len*wid} \end{bmatrix}$$

Let n be the width of weight, where $n < wid$. For the 1st to $(l/C - 1)$th submatrices z_i, the $(n+1)$th to (wid)th elements of each row are filled with zero. For the (l/C)th matrix, the $(n+1)$th to (wid)th elements of the 1st to $(l\%C)$th rows are filled with zero. The$(l\%C+1)$th to (C)th rows are filled with zero. For the input data x, pack to the first n positions of the submatrix.

3.4 Horizontal SIMD

In the Homo-ELM scheme, the author encrypts the data features and network weights one by one, which requires a large number of homomorphic operations. For example, the user needs to predict an image with 64 pixels, so he needs to call the encryption method in the BGV scheme 64 times for a single image, and the obtained 64 polynomials also need to do a lot of homomorphic operations with the connection weight w. Although the Chinese remainder theorem can be used to package pixels in the same location, in practical applications, the user often does not perform too many prediction requests in the same period, so it is more practical to reduce the latency of inference stage.

3.4.1 Rotation and Summation

In CKKS scheme, by using Frobenius automorphism mapping, the ciphertext $m(x)$ corresponding to the original private key s_2 can be converted into the ciphertext $m(x^{5^i})$ corresponding to the private key s_1 by switching key swk, then the corresponding ciphertext slot space will rotate i positions to the left, and the plaintext value corresponding to the ciphertext slot changes from the original $\{x_1, x_2, \cdots, x_{N/2}\}$to

Table 1: Polynomial approximation result of TanhRe

	Result
$tr_3\ (x)$	$-0.0068661x^3 + 0.0828685x^2 + 0.7495332x + 0.1245075$
$tr_7\ (x)$	$-0.0000419x^7 + 0.0000704x^6 + 0.0022615x^5 - 0.0044224x^4$ $-0.0422012x^3 + 0.1474116x^2 + 0.8906998x - 0.0137859$

$\{x_i, x_{i+1}, \cdots, x_{N/2}, x_1, \cdots, x_{i-1}\}$, which is called rotation. For the encoded matrix vector, the method of rotation summation can be used to calculate the sum of rows or columns quickly.

Algorithm 1 Colsum

Input: A ciphertext c, an encryption of $len * wid$ matrix satisfying len is the power of 2
Output: A ciphertext $c^{'}$
$c^{'} = c$;
for $1 \leq i \leq \log_2 len$
  $cRot = Rotate(c, (-1) * 2^{i-1} * wid)$;
  $c^{'} = Add(c^{'}, cRot)$;
end for
return c

Algorithm 2 Rowsum

Input: A ciphertext c, an encryption of $len * wid$ matrix satisfying wid is the power of 2
Output: A ciphertext $c^{'}$
$c^{'} = c$;
for $1 \leq i \leq \log_2 wid$
  $cRot = Rotate(c, 2^{i-1})$;
  $c^{'} = Add(c^{'}, cRot)$;
end for
return c

3.4.2 Matrix-Vector Multiplication in Ciphertext

To obtain the prediction results for a single data, it is necessary to multiply between the weight matrix and the intermediate vector. Firstly, the column summation algorithm is needed for the data ciphertext vector x, and the x needs to be copied and filled. The matrix ciphertext is then multiplied by the data ciphertext x, and the row summation algorithm is carried out. Finally, the first column of the matrix corresponding to the ciphertext is the multiplication result of matrix and vector.

3.4.3 Multi-hidden Node Calculation

When the parameter N in the CKKS scheme is set, the capacity of a ciphertext is fixed. When ELM is applied for complex classification tasks, ELM may need a large number of hidden nodes, and multiple ciphertext matrices are required to pack all the weights for calculation.

Algorithm 3 MatrixMulVector

Input: A matrix ciphertext c, a vector ciphertext v
Output: A ciphertext $c^{'}$
$v^{'} = Colsum(v)$;
$tmp = Mult(c, v^{'})$
$tmp^{'} = Rescale(tmp)$
$c^{'} = Rowsum(tmp)$;
return $c^{'}$

Algorithm 4 Multi-hidden Node Calculation

Input: A ciphertext c, an encryption of $len * wid$ matrix satisfying len is the power of 2
Output: A ciphertext $c^{'}$
$f = \emptyset$;
for $1 \leq i \leq n$
  $c^{'} = MatrixMulVector(c_i, v)$;
  $f = Add(f, c^{'})$
end for
return f

4 Construction of Security Model

In the previous scheme based on Paillier homomorphic encryption, the CPP-ELM and the PP-ELM schemes retrieve the initial connection weights from the third-party server to the local and then use their own data set for operation and encryption. Finally, they return to the third-party server for integration, and calculate the weights from the hidden layer to the output layer. When the user predicts, the user directly obtains the trained model parameters and processes data locally. Although this protects user's data privacy and security, it exposes the entire ELM model. After the user predicts the result once, subsequent cloud services are no longer needed. For commercial service providers, this is not willing to see.

In the previous HOMO-ELM scheme, the model parameters are encrypted with a public key sp_{pk} and uploaded to the cloud server. When the user needs data prediction, the data is encrypted with sp_{pk} and uploaded to the cloud server for calculation. Finally, retrieve the calculated data locally and decrypt it with the user's private key to get the result. Although the HOMO-ELM ensures that neither the training model nor user information is disclosed, since the model in the cloud server is encrypted and uploaded by the model provider, other users who want to use it need to send their data encrypted with sp_{pk} to the server but cannot rule out malicious theft by the service provider. On the other hand, even if the

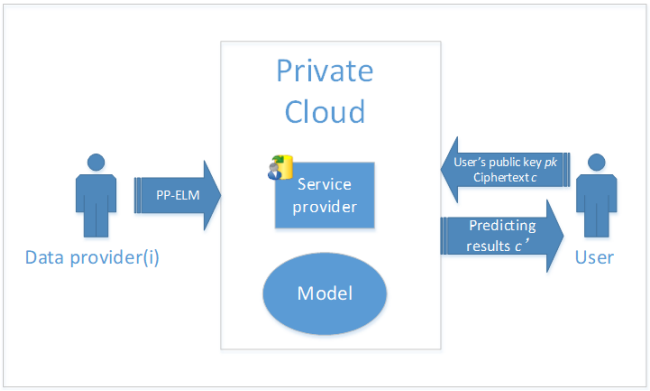


Figure 2: Operation process of CKKS-ELM

service provider does not steal user data, there is still a risk of eavesdropping when the plaintext returns to the user.

This paper proposes a privacy preserving system with better security and more suitable for commercialization. In the model training phase, we follow the PP-ELM scheme modified in Section 3.1 for model training, so that the data provider does not disclose any data, but the service provider can use the data for model training. In terms of prediction, the user generates his public and private key locally, encrypts the data with the public key, and then sends the public key and the encrypted data to the service provider. It is worth noting here that the user only needs to send the public key at the time of the first prediction, and there is no need to send the public key in later use. Then the service provider performs homomorphic operation on the encrypted data by using the ELM model and returns the encrypted data to the user, and the user decrypts the ciphertext with his private key to obtain the prediction result. Throughout the process, the data provider will not leak any data, the service provider's model will not be stolen, and neither the user's data nor the prediction results will be leaked. The process is shown in Figure 2 and the scheme comparison is shown in Table 2.

4.1 Prediction Process

4.1.1 Normalization of Data

Normalization can compress the data to the specified interval, simplify the neural network calculation and improve the classification performance. In general, the model training bases on normalized data, so the input data need to be normalized in data prediction. For the input data x, normalize it to the interval $(-1, 1)$ and do the following equation:

$$x_{Normalization} = 2 \cdot \frac{x - min(x)}{max(x) - min(x)} - 1.$$

Where $min(x)$ and $max(x)$ represent the minimum and maximum values of a feature of x, recorded by the service provider while training the model.

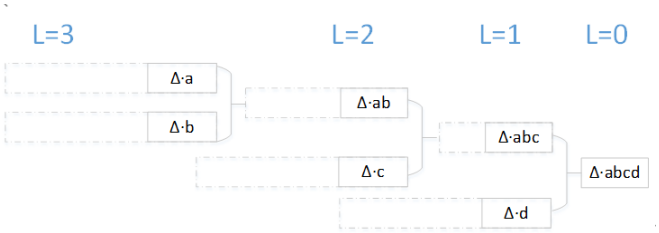


Figure 3: Multiplication in sequence

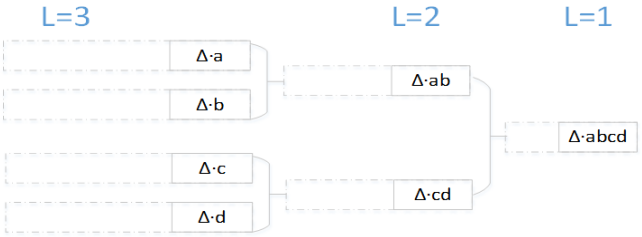


Figure 4: Multiplication in optimal sequence

4.1.2 Level Optimization

Suppose the degree of the polynomial is k. If the homomorphic multiplication operation is performed in sequence, the number of levels L that the highest degree term needs to consume is k. When the low-order term and the high-order term are added and merged, they are aligned with the lowest-order ciphertext, so the number of layers consumed by the entire polynomial operation is k. However, you can reduce the level of consumption through level optimization, and reduce the number of layers consumed from k to $\log_2 k + 1$, as shown in the Figure 3 and Figure 4:

4.1.3 Procedure

In this section, we integrate the above-mentioned to describe the complete process of data prediction phase. The concrete procedure will be given in Algorithm 5 and Algorithm 6.

5 Experiments

5.1 Experimental Setup

The experiments were carried out from the Dell computer, which has a core i5 8500 3.00 GHz CPU, 32 GB RAM, and based on Matlab 2018b and homomorphic encryption library SEAL. The main hyperparameters of SEAL are as follow: parameter $N = 16384$ of polynomial ring, levels $L = 7$, $scale = 2^{32}$, the length of q is 324bit and error standard deviation $\sigma \approx 3.2$. According to the running results of the LWE-estimator proposed in APS15 scheme [2], the ciphertext generated under the above parameter setting method meets the 128-bit security and achieve the 128-bit security against a quantum computer.

Table 2: Security comparison with three schemes

Scheme	Model protection	User data privacy protection	Multi-user safety use
PP-ELM	×	✓	✓
CPP-ELM	×	✓	✓
HOMO-ELM	✓	✓	×
Our scheme	✓	✓	✓

Algorithm 5 User Prediction

Input: A data $x \in \mathbb{R}^{feature}$
Output: Prediction classification results *predict*
//User initialize the CKKS scheme in local environment, set polynomial degree N, module Q and scale factor p, and generate public key pk, private key sk, switching key swk.
$scheme = init(params)$;
//read in data,$x = x_1, x_2, \cdots, x_{feature}$
$x = getData()$;
//encrypt x into ciphertext with pk
$ct = enc(x, pk, p)$;
if the user has not predicted or needs to change the public key **then**
  //Send ct, pk, swk and p to the service provider
  $result = ServerProcessing(ct, pk, swk)$;
else
  //Send ct to the service provider
  $result = ServerProcessing(ct)$;
end if
// Retrieve the result and decrypt the ciphertext with user's private key
$p_{res} = dec(result, sk, p)$;
// For the data with k labels, the user finds the maximum value in the first k data slots in the plaintext p_res. The corresponding index idx is the current prediction result.
$predict = MaxIn(p_{res})$;
return *predict*

Algorithm 6 ServerProcessing

Input: A ciphertext $x^{'}$,public key pk, switch key swk, and scale factor p
Output:A ciphertext $c^{'}$ with prediction result
//Gets $min(x)$ and $max(x)$ from model training and normalizes $x^{'}$
$ct = normalize(x^{'})$;
//duplicate ct vertically
$ct = colsum(ct)$;
//Get the weight w and put it into the vector according to the data encoding format, and then generate the corresponding plaintext polynomial. Finally, multiply with ct and perform a rescale operation
$IW_X = Rescale(CMult(getWeightIW(), pk, p)))$;
//duplicate IW_X horizontally
$IW_X = Rowsum(IW_X)$;
//Do an addition between IW_X and bias B
$IW_{XB} = Add(IW_X, enc(getWeightB()), pk, p))$;
//Approximate polynomial activation is performed, and H_{out} is the output of the hidden layer.
$H_{out} = poly\_activate(IW_{XB})$;
// Except for the useful information in the first column of H_{out}, the other useless information needs to be eliminated by multiplying a mask vector with H_{out}.
$H_{out} = Rescale(CMult(getMaskVec(), H_{out})))$;
//The weight LW from the hidden layer to the output layer is multiplied by H_{out} and rescaled
$LW_H = Rescale(CMult(H_{out}, getWeightLW\ ()))$;
//Finally, perform a Colsum operation to get the final result
$result = colsum(LW_H)$;
return *result*

To objectively evaluate the classification performance of the model, the data set will be randomly shuffled 10 times, and the 5-cross validation method will be used for each shuffled data set. Since the input layer's connection weights and the hidden layer in the ELM are randomly generated, which will cause the randomness of the model accuracy, each evaluation is carried out 50 times, and the average value is taken. Finally, the average value of all experimental results of the activation function is taken.

5.2 Experimental Results and Comparison

To verify the classification performance of our proposed scheme, five real-world datasets from the UCI Machine Learning Repository are used for classification prediction: Iris Data Set, Glass Identification Data Set, Ionosphere Data Set, Skin Segmentation Data Set and Landsat Satellite Data Set. The information about these datasets is shown in Table 3:

We compare the classification performance of the activation function proposed in HOMO-ELM. The first four lines of the Table 4 below are the experimental results of the model accuracy, all of which are predicted on encrypted data. The numbers in the table below represent the number of hidden nodes. $tr_3(x)$ and $tr_7(x)$ are defined in the Table 1. The activation functions proposed in HOMO-ELM with better classification performance are $x_2 = x^2 + x + 1$ and $x_3 = x^3$.

Experiments show that TanhRe activation function has better model performance than HOMO-ELM in the data

Table 3: Datasets

Datasets	**Total**	**Features**	**Classes**
Iris	150	4	3
Glass	214	9	6
Ionosphere	351	34	2
Satellite	6435	36	6
Skin	245057	3	2

Table 4: Classification accuracy

Method \ Datasets	$Iris_{100}$	$Ionosphere_{50}$	$Glass_{100}$	$Satellite_{100}$	$Satellite_{300}$	$Satellite_{500}$	$Skin_{100}$
x_2	96.5%	87.1%	64.34%	85.29%	86.14%	86.29%	95.86%
x_3	95.58%	83.09%	64.73%	84.24%	86.67%	87.36%	97.09%
tr_3	96.38%	88.2%	66.89%	85.49%	87.06%	88.7%	97.15%
tr_7	97.65%	87.08%	66.88%	86.43%	87.71%	89.43%	98.53%
HOMO-ELM	89.55%	-	-	78.62%	-	-	-
PP-ELM	-	-	65.4%	85%	87.5%	-	-

set with fewer features. Because CKKS scheme naturally supports the characteristic of decimal, it can approximate any nonlinear activation function for different data sets to obtain better classification accuracy, while HOMO-ELM is limited in expressing activation function because of the integer-based BGV scheme.

Let the number of hidden nodes be L, the number of data features be F, the number of classes be K and the highest degree of activation functions as T. From the perspective of efficiency, the complexity of homomorphic operations for each data prediction by HOMO-ELM is $\mathcal{O}(LF + KF + \log_2 T)$ and the number of ciphertexts for communication is $F + K$. Let the highest degree of the polynomial ring of CKKS is N, the complexity of homomorphic operations for each data prediction in our scheme is $\mathcal{O}(\frac{LF+KF}{N} + \log_2 T)$ and the number of ciphertexts for communication is $\lceil \frac{F}{N} \rceil + \lceil \frac{K}{N} \rceil$, which significantly reduced latency for user prediction. Table 5 summarizes benchmark results of our experiments.

For more comparisons, the accuracy of skin data set in PP-LG results from the classification test of Logistic model under Privacy Protection proposed by Aono *et al.* [3] is 93.89%. F. Al-Obeidat *et al.* [1] achieved 88.29% accuracy on their proposed optimized method while achieving 82% and 85.71% on ID3 and C4.5 respectively. DU *et al.* [8] proposed a privacy preserving optimized k-NN algorithm and the highest accuracy of Iris and skin segmentation dataset is 97% and 98%, which is closed to our scheme on tr_7 method. But they need a few seconds to interfere with a single input. Sarpatwar *et al.* [23] proposed a systematic approach to perform multiplicative depth constrained knowledge distillation that enables efficient encrypted inference. They carried out on two other UCI Dataset: Bank Management and MAGIC Gamma Telescope, and we achieve a closed result in accuracy but very low latency since their SIMD strategy need lots of computation consumption, which leads to approximately 5 minutes. The above facts all show that the privacy preserving extreme learning machine scheme proposed in this paper not only guarantees certain accuracy in the lightweight privacy preserving model but also has lower computation cost.

6 Conclusions

In this paper, a new privacy preserving extreme learning machine scheme called CKKS-ELM is proposed. In the aspect of homomorphic encryption, we use the CKKS scheme, which is more suitable for machine learning applications. In the aspect of ELM, we propose horizontal SIMD processing method and better activation function. In communication construction, we propose a privacy preserving system that is more suitable for commercialization. Extreme learning machine has been studied as a classifier with good classification performance. The future research direction can combine fully homomorphic encryption with the new extreme learning machine theory while taking into account efficiency and accuracy.

Acknowledgments

This study was supported by the National Natural Science Foundation of China (No.62062009), the Science and Technology Major Project of Guangxi of China (AA17204058-17) and the Science and Technology Major Project of Guangxi of China (Guike AA18118047-7). The authors gratefully acknowledge the anonymous reviewers for their valuable comments.

Table 5: Runtime in different dataset and setting (Seconds)

Method \ Datasets	$Iris_{100}$	$Ionosphere_{50}$	$Glass_{100}$	$Satellite_{100}$	$Satellite_{300}$	$Satellite_{500}$	$Skin_{100}$
x_2	0.126	0.16	0.143	0.164	0.391	0.505	0.126
x_3	0.151	0.2	0.18	0.211	0.525	0.685	0.15
tr_3	0.159	0.22	0.187	0.217	0.54	0.706	0.157
tr_7	0.322	0.257	0.291	0.322	0.843	1.128	0.245

References

[1] F. Al-Obeidat, A. Al-Taani, N. Belacel, L. Feltrin, and N. Banerjee, "A fuzzy decision tree for processing satellite images and landsat data," *Procedia Computer Science*, vol. 52, pp. 1192–1197, 2015.

[2] M. Albrecht, R. Player, and S. Scott, "On the concrete hardness of learning with errors," *Journal of Mathematical Cryptology*, vol. 9, no. 3, pp. 169–203, 2015.

[3] Y. Aono, T. Hayashi, L. Phong, and L. Wang, "Privacy-preserving logistic regression with distributed data sources via homomorphic encryption," *IEICE TRANSACTIONS on Information and Systems*, vol. 99, no. 8, pp. 2079–2089, 2016.

[4] Z. Cao, L. Liu, and Y. Li, "Ruminations on fully homomorphic encryption in client-server computing scenario," *International Journal of Electronics and Information Engineering*, vol. 8, no. 1, pp. 32–39, 2018.

[5] F. Çatak and A. F. Mustacoglu, "CPP-ELM: Cryptographically privacy-preserving extreme learning machine for cloud systems," *International Journal of Computational Intelligence Systems*, vol. 11, no. 1, pp. 33–44, 2018.

[6] J. H. Cheon, K. Han, A. Kim, M. Kim, and Y. Song, "A full RNS variant of approximate homomorphic encryption," in *International Conference on Selected Areas in Cryptography*, pp. 347–368. Springer, 2018.

[7] J. H. Cheon, A. Kim, M. Kim, and Y. Song, "Homomorphic encryption for arithmetic of approximate numbers," in *International Conference on the Theory and Application of Cryptology and Information Security*, pp. 409–437. Springer, 2017.

[8] J. Du and F. Bian, "A privacy-preserving and efficient k-nearest neighbor query and classification scheme based on k-dimensional tree for outsourced data," *IEEE Access*, vol. 8, pp. 69333–69345, 2020.

[9] C. Gentry, "Fully homomorphic encryption using ideal lattices," in *Proceedings of the forty-first annual ACM symposium on Theory of computing*, pp. 169–178, 2009.

[10] M. Hashimoto and Q. Zhao, "An ELM-based privacy preserving protocol for implementing aware agents," in *2017 3rd IEEE international conference on cybernetics (CYBCONF)*, pp. 1–6. IEEE, 2017.

[11] M. Hashimoto and Q. Zhao, "A privacy preserving protocol for cloud-based implementation of aware agents," *Journal of Information Processing*, vol. 26, pp. 486–496, 2018.

[12] G. B. Huang, H. Zhou, X. Ding, and R. Zhang, "Extreme learning machine for regression and multiclass classification," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 42, no. 2, pp. 513–529, 2011.

[13] G. B. Huang, Q. Y. Zhu, and C. K. Siew, "Extreme learning machine: theory and applications," *Neurocomputing*, vol. 70, no. 1-3, pp. 489–501, 2006.

[14] Y. Jing. *Research on Key Technologies of Homomorphic Encryption*. PhD thesis, University of Electronic Science and Technology of China, 2019.

[15] S. Kuri, T. Hayashi, T. Omori, S. Ozawa, Y. Aono, L. Wang, S. Moriai, et al., "Privacy preserving extreme learning machine using additively homomorphic encryption," in *2017 IEEE symposium series on computational intelligence (SSCI)*, pp. 1–8. IEEE, 2017.

[16] J. Lin, J. Yin, X. Zhang, Z. Cai, and Y. Ming, "Optimization mechanism for secure outsourcing extreme learning machine in cloud computing," *Computer & Digital Engineering*, vol. 47, no. 1, pp. 157–160, 2019.

[17] L. Liu and Z. Cao, "Analysis of two confidentialitypreserving image search schemes based on additive homomorphic encryption," *International Journal of Electronics and Information Engineering*, vol. 5, no. 1, pp. 1–5, 2016.

[18] Y. Ma, L. Wu, X. Gu, J. He, and Z. Yang, "A secure face-verification scheme based on homomorphic encryption and deep neural networks," *IEEE Access*, vol. 5, pp. 16532–16538, 2017.

[19] M. Maimaitiyiming, V. Sagan, P. Sidike, and M. T. Kwasniewski, "Dual activation function-based extreme learning machine (elm) for estimating grapevine berry yield and quality," *Remote Sensing*, vol. 11, no. 7, p. 740, 2019.

[20] D. E. Ratnawati, Marjono, Widodo, and S. Anam, "Comparison of activation function on extreme learning machine (elm) performance for classifying the active compound," in *AIP Conference Proceedings*, vol. 2264, p. 140001. AIP Publishing LLC, 2020.

[21] R. L. Rivest, L. Adleman, M. L. Dertouzos, et al., "On data banks and privacy homomorphisms," *Foundations of secure computation*, vol. 4, no. 11, pp. 169–180, 1978.

[22] S. Samet and A. Miri, "Privacy-preserving back-propagation and extreme learning machine algo-

rithms," *Data & Knowledge Engineering*, vol. 79, pp. 40–61, 2012.

[23] K. Sarpatwar, K. Nandakumar, N. Ratha, J. Rayfield, K. Shanmugam, S. Pankanti, and R. Vaculin, "Efficient encrypted inference on ensembles of decision trees," *arXiv preprint arXiv:2103.03411*, 2021.

[24] W. Wang, Y. Gan, C. M. Vong, and C. Chen, "Homo-elm: fully homomorphic extreme learning machine," *International Journal of Machine Learning and Cybernetics*, pp. 1–10, 2020.

Biography

Kunhong Li is a master degree student in the school of Computer and Electronic Information, Guangxi University. His research interests focus on information security.

Ruwei Huang received her Ph.D from Xi'an Jiaotong University in 2012. She is currently an associate professor of the school of Computer and Electronic Information, Guangxi University. Her research interests include cloud computing and fully homomorphic cryptography.

The System Adoption Evaluation of RFID Safety Management System on Campus

Yu-Ying Hsieh[1], Liang-Hsu Chang[2], Anthony Y. H. Liao[3], Cheng-Ying Yang[4], and Min-Shiang Hwang[1,5]

(Corresponding author: Min-Shiang Hwang)

Department of Computer Science and Information Engineering, Asia University[1]
No. 500, Lioufeng Rd., Taichung 41354, Taiwan (R.O.C.)
Department of Management Information Systems, National Chung Hsing University[2]
Taichung 403, Taiwan (R.O.C.)
Department of M-Commerce and Multimedia Applications, Asia University[3]
Department of Computer Science, University of Taipei[4]
Taipei 10048, Taiwan (R.O.C.)
Department of Medical Research, China Medical University Hospital, China Medical University[5]
No. 91, Xueshi Rd., Taichung 40402, Taiwan (R.O.C.)
Email: mshwang@asia.edu.tw

(Received July 10, 2020; Revised and Accepted Dec. 20, 2021; First Online Dec. 31, 2021)

Abstract

In this article, we propose a campus safety management system based on radio frequency identification (RFID) technology. In this work, the evaluation based on the system adoption was used to establish a campus-safety management system model and related implementation issues. With the implementation of the RFID system for campus safety management, the system could save the administrators' time and improve the efficiency of service quality. In addition, it helps to prevent the occurrence of security incidents on campus. The critical success factors of the system include the participation of students and parents. Finally, the evaluation on the system adoption is made by questioners. The results show the first three factors affecting the administrators in the school to decide whether the system is used the convenience of use, agreement of students' parents, and the relative advantage. However, school competition and school reorganization could not be the preferred factors to adopt the system.

Keywords: Campus Safety Management; RFID; System Adoption Evaluation

1 Introduction

School provides the primary function of teaching and learning [9]. However, the school might not be a safe but public environment for the people walking in. For example, the restrooms, the basements, the rooftops, the stairways, the parking lots, and the construction locations without proper insulation might be where the accidents are held [3, 5]. Especially in the school, the staffs have to pay more attention to those students with physical disabilities [3]. The Campus Security Report indicated that, in addition to teaching and learning services, a school must have a safe and reliable campus environment [15]. Hence, campus safety is the most critical concern for the school administration. Traditionally, campus safety management is primarily based on human resources and auxiliary equipment, such as surveillance cameras. Students with particular disabilities cannot respond to the dynamic changing circumstances immediately. Therefore, it would be harmful and cause various dangers. Under the limitation of human resources in the school, we might complement the increasing works on the safety guards with the assistance of information technology [2].

Innovation is defined as something an individual or receiver considers to be a new concept or behavior [7]. Diffusion is a process in which the innovation is expanded and spread through specific ways to individuals or organizations in the social system and, then, accepted by members in the social system. In addition to consideration on adoption by organizations, diffusion of innovative technology is significantly related to the levels of acceptance and fitness at the end-users toward technology [8]. For example, radio frequency identification (RFID) technology [4, 6] uses radio waves to carry the identification information for recognization. The administration might employ this technology at the school for a safe campus. RFID provides real-time information to the administra-

tion. This information could rescue the accident on the campus in a short time. Moreover, the integrated system could provide some safety management schemes and construct a safe campus to reduce possible safety incidents [1, 10]. Therefore, it might benefit the school management.

To approach a safe campus by employing RFID, the RFID safety system in this work refers to the safety management for supervising the hazardous areas on the campus [4]. The information technology used consists of the hardware and the software. The hardware includes the active tags and the RFID reader. The integrated framework for the safety campus is constructed with the connection among these elements, including the hardware and software management. Initially, the system considers the scenario as the following. Students wear active tags on their wrists on the campus, and the RFID readers are located in various dangerous areas. The RFID safety system will have an alarm signal when the RFID reader receives the signal sent by the active tag. Furthermore, it automatically sends a text or e-mail related to the safety management of hazardous areas on campus and informs the administrators for safety purposes. Besides, based on the innovation property, the system is evaluated with relative advantage, compatibility, ease of use, privacy risk, and cost. The following section describes the proposed system considering the innovation properties. Section III shows the proposed system adoption evaluation according to the census results. Finally, the conclusion is given in the final.

This paper is organized as follows. First, in Section 2, we introduce the consideration of the proposed system. Then, the system adoption evaluation is described in Section 3. Finally, a conclusion is conducted in Section 4.

2 System Consideration

RFID, the e-tag, uses radio waves for transmitting identification information. An RFID system consists of an RFID tag, an RFID reader, and the application [11–14]. The RFID system and safety management on campus refer to the dangerous area management service in three disabled students. The project got permission from the students' parents for the human rights concern. All students in this project wear RFID active tags on their wrists. On the campus, the RFID readers are installed in various dangerous areas.

The system is designed with the following scenario. When the RFID reader receives the signal, the safety management system will alarm and send text or e-mail regarding safety to the administrators on campus. It means the students are close to those hazardous areas. For the safety concerns, all administrators should pay attention to safeguarding the students. Mainly, the disabled students might not concentrate their attention on the environment changing rapidly, and they hardly respond to avoid the danger in a short time. Hence, this system helps the staff to avoid accidents efficiently.

Beyond considering system functions, the system is established with other factors like the following. In Taiwan, a student ID card with RFID is already used on campus for library use and making payments. Hence, the degree of acceptance for the students to use the RFID is high, without a doubt. It is the relative advantage of adopting RFID in the safety system. Besides, danger detection is necessary for students to prevent danger. Therefore, using the RFID for danger detection could be accepted for the students who currently use this technology. The compatibility of RFID includes the function of student identification and payment convenience and provides danger prediction and accident reduction. Hence, the RFID safety system could be realized and could be extended. Moreover, RFID is a device for easy use. For example, students could enter the library if they carry an RFID card. Moreover, almost all stores accept payment with RFID cards. Therefore, it gives people an easy life. Hence, almost all people could appreciate the extra benefits of using RFID. However, the tag used in this project is active. Therefore, some users might worry about their health and privacy.

3 System Adoption Evaluation

Traditionally, safety management primarily works for the administration office in the school. The performance depends on the human resource, including safety guards and administrators. However, the human resource in the school is limited. Hence, employing information technology is an efficient method. The information technology for safety management on campus could be innovative and could be applied to almost all schools.

According to Innovation diffusion theory by Rogers, innovation is defined as something an individual considers to be a new concept or behavior. Diffusion is a process in which an innovation is expanded and spread through particular communication channels to individuals or organizations in the social system and, then, accepted by those people. Innovation diffusion mainly depicts specific special processing. By the way, the individuals or organizations could understand and adopt the innovation for some purposes [7]. In addition to consideration on adoption by organizations, diffusion of innovative technology is significantly related to levels of acceptance and fitness at the clients.

Wireless application with RFID is convenient to use. Primarily, it is employed in the safety management system on the campus. Hence, the system evaluation could be based on personal behavior factors. There is compatibility, the convenience of use, private risk, and cost. Compatibility is the degree that the RFID safety management system is consistent with the user's current needs. It will be affected by the users' knowledge of the operating experience. Also, it will affect the adoption in the further. Convenience is the degree that end-users could easily use the RFID safety management system. The lower degree

represents the users without a desire to use the system. Once the system could not significantly help the users improve the existing difficulty, the users could not adopt the system anymore. Privacy risk is an essential factor for users who recognize the risk it might bring by using the system. Since there are actual laws and related policies to regulate the tag reading of RFID, some risks might come with this proposed system. The worry about this risk might prevent the system adoption. Another factor could be the cost of the system. Cost in this safety management system includes the hardware and software. Although the system effectively reduces the cost of human resources, not only the building, end-users could pay the maintenance cost if the system is working. Therefore, based on the above consideration, the evaluation should reflect the following aspects.

Q1. Relative advantage;

Q2. Convenience of use;

Q3. Compatibility;

Q4. Privacy risk;

Q5. Cost.

On the other hand, some external factors could affect system adoption. For example, the support of the safe management system will affect the adoption decision by the users. Hence, the system evaluation should consider some factors from the system suppliers. In the integrated RFID system, the establishment, cooperating service, operation, and problem-solving support provided by software and hardware establishing firms will affect the adoption decision of the introduced RFID campus safety system. Besides, the support from the government could be a factor to affect the system adoption by the school. Usually, the Ministry of Education is the primary support to the school to use the safe management scheme. It is related to the critical policy for all the schools. With this support, the system could be accepted by the schools.

Furthermore, in the RFID safe system, students should wear the tag; as mentioned above, human rights should obtain consent from the students' parents. Hence, the agreement could be one evaluation factor. Moreover, the application of RFID could be an attractive point for the school could be another evaluation factor. With this IT application, the school might give a good impression to all students, and it increases the school competition and, then, the school has a solid willingness to use this system. Based on the above consideration, the evaluation should add the following aspects.

Q6. Support of software and hardware suppliers;

Q7. Government support;

Q8. Agreement of parents;

Q9. School competition.

Moreover, some internal factors in the school could be evaluated as the impact factors. For example, disabled students could not respond to changing circumstances and determine various dangers, and a massive amount of human resources is required to assist in students' safety on campus. Hence, the school recognizes that using this safe management system could help students' safety. If it is positive, the school will choose this system for safe management. Also, if a disaster prevention and rescue scheme exists within the administration policy, the school might adopt the RFID safe management system. As the school is responsible for many students, it often employs quick and convenient schemes to maintain safety. If the system is a requirement to the school, administrators could adopt the system without a doubt. On the other hand, on the campus, the staff could affect the system's adoption. Almost all staff should prepare to handle the accident immediately. Thus, the support from staff, including the administrators and teachers, could be the factors that affect the system adoption. Hence, the evaluation should include the following opinions.

Q10. School reorganization;

Q11. Requirement of safety management;

Q12. Support from administrative executives;

Q13. Support from teachers.

This work uses the questionnaire to evaluate the RFID safe management system on campus. The purpose is to explore the practical case of schools that adopt RFID systems for those disabled students. According to the consideration above, the decision to accept and use this system depends on the user. It includes reasoned action theory, planned behavior theory, and the technology acceptance model (TAM). Also, this work combines the innovation diffusion theory and the theory of planned behavior, as shown in Figure 1.

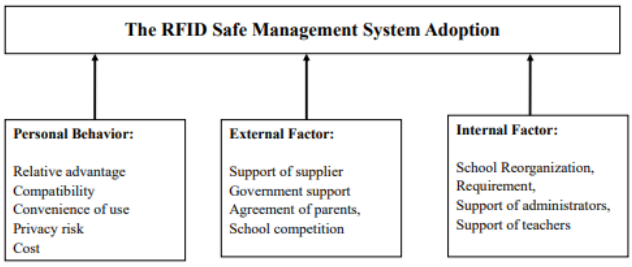


Figure 1: The consideration to adopt RFID safe system

This work collects 30 valid samples to evaluate the system adoption. According to Likert scaling, each question is scaled from 1 to 5 points to represent strongly disagree, disagree, ordinary, agree, and strongly agree, respectively. In the questionnaire, the "School's adoption of RFID system for managing safety in the dangerous areas on campus" is divided into three aspects, including "innovation

Table 1: The statistical analysis of the consideration factors to adopt the system

	Mean	Standard Deviation	Standard Error	Order
Relative advantage	4.37	.490	.089	3
Cost	4.17	.592	.108	8
Convenience of use	4.47	.629	.115	1
Compatibility	4.17	.874	.160	8
Privacy risk	4.27	.640	.117	5
Support of supplier	4.30	.651	.119	4
Government support	4.17	.648	.118	8
Agreement of parents	4.40	.563	.103	2
School competition	3.80	.887	.162	13
School reorganization	3.90	.712	.130	12
Requirement of safety management	4.20	.610	.111	7
Support of administrators	4.17	.747	.136	8
Support of teachers	4.27	.691	.126	5

property", "special education school environment," and "organization property". A five-point scale is used to divide the evaluation of senior school managers or administrative officers toward factors affecting the application of the system on campus safety in the dangerous areas into: "strongly agree", "agree", "ordinary", "disagree" and "strongly disagree". The lowest value, 3.5 of the median value (M), is used as the entire evaluation of particular education school senior managers or administrative officers toward factors affecting the application of RFID system on campus safety in dangerous areas. The higher the score indicates, the more significant the impact of a senior manager or administration officer on applying the RFID system.

The results show the average and standard deviation of the 13 factors of the three aspects of an RFID system in managing safety in the dangerous areas on camps are as shown in Table 1. Factors of school's senior manager or administration officer affecting the introduction of RFID system for managing safety in the dangerous areas on campus. The top is "Convenience of use" (M=4.47), followed by "Agreement of parents" (M=4.40) and "relative advantage" (M=4.37). It indicates that the impact of senior managers or administrative officers toward the introduction of an RFID system in the management of dangerous areas on campus is most significant on "Convenience of use,"; but the impact is minor on "School competition" (M=3.80) and "School organization" (M=3.90).

According to the above results, when the RFID system is introduced in a school for disabled students, most senior managers or administrators believe that the RFID safety management system is effectively helpful and valuable to those students without rapid reaction. In addition, the RFID helps to give a safe environment on campus significantly.

4 Conclusion

This work uses a questionnaire to evaluate the RFID safe management system on campus. Although the sample is limited, the survey results show some important information. It discovers that radio frequency identification (RFID) could be a noticeable trend for safe management on campus. The first three factors affect the administrators in the school to decide whether the system is used the convenience of use, the agreement of students' parents, and the relative advantage. However, school competition and school reorganization could not be the preferred factors to adopt the system. It might be the noble application could not be accepted for the people who are not familiar with information technology. Some efforts could continue to promote this RFID safe management system in the future. For example, the RFID campus safety management system is currently used for the researcher's proposal. Hence, there is no regulation to describe the security requirement. It might cause the worry of the users. We should consider the legislation on the reading of RFID tags before the application is widespread.

Acknowledgments

The Ministry of Science and Technology partially supported this research, Taiwan (ROC), under contract no.: MOST 109-2221-E-468-011-MY3, MOST 108-2410-H-468-023, and MOST 108-2622-8-468-001-TM1.

References

[1] 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.

[2] Y. Y. Chen, and M. L. Tsai, "An RFID solution for enhancing inpatient medication safety with real-time verifiable grouping-proof," *International Journal of Medical Informatics*, vol. 83, no. 1, pp. 70–81, 2014.
[3] A. G. Dell, D. A. Newton, and J. G. Petroff, *Assistive Technology in the Classroom: Enhancing the School Experiences of Students with Disabilities*, (2nd Edition), Pearson, 2016.
[4] H. Fernandes, V. Filipe, P. Costa, and J. Barroso, "Location based services for the blind supported by RFID technology," *Procedia Computer Science*, vol. 27, pp. 2–8, 2014.
[5] J. Hardwick, *A Safe Space for Dangerous Ideas; A Dangerous Space for Safe Thinking*, Aug. 13, 2014. (`https://hybridpedagogy.org/safe-space-dangerous-ideas-dangerous-space-safe-thinking/`)
[6] Y. Liu, A. Levitt, C. Kara, C. Sahin, G. Dion, and K. R. Dandekar, "An improved design of wearable strain sensor based on knitted RFID technology," in *Proceedings of IEEE conference on Antenna Measurements & Applications (CAMA'16)*, pp. 1-4, 2016.
[7] E. V. Rogers, *Diffusion of Innovations*, 4th Edition, the Free Press, 1995.
[8] J. Y. L. Thong, "An integrated model of information systems adoption in small business," *Journal of Management Information Systems*, vol. 15, no. 4, pp. 187–214, 2015.
[9] M. Weber, L. Wagner, and W. Ruch, "Positive feelings at school: on the relationships between students' character strengths, school-related affect, and school functioning," *Journal of Happiness Studies*, vol. 17, no. 1, pp. 341–355, 2016.
[10] 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.
[11] C. H. Wei, M. S. Hwang, Augustin 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.
[12] 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.
[13] 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.
[14] C. H. Wei, M. S. Hwang, Augustin Y. H. Chin, "Security analysis of an enhanced mobile agent device for RFID privacy protection," *IETE Technical Review*, vol. 32, no. 3, pp. 183–187, 2015.
[15] Wellesley College Campus Police, *Wellesley College Annual Campus Security and Fire Safety Report 2021*, Dec. 31, 2021. (`http://www.wellesley.edu/police/securityrpt#ejJxRy7YxPmx5GXW.97`)

Biography

Yu-Ying Hsieh received the Bachelor of Education in National Taiwan Normal University in 1983 and received an M.S. of Social Welfare in Providence University, Taiwan, in 2003. She served as a kindergarten teacher and principal from 1982 to 2005. She has repeatedly won the Kindergarten Model Award and Social Welfare Service Award issued by government agencies from 1996 to 2003. She was currently studying for a Ph.D. in the Department of Computer Science and Information Engineering of Asia University. Her research interests include applying information technology in-game and event design, etc.

Anthony Y. H. Liao received his M.S. degree in computer science and the Ph.D. degree in computer science and engineering both from the University of Louisville, Louisville, Kentucky, USA. He is currently an associate professor and the Chairman of the Department of M-Commerce and Multimedia Applications, Asia University, Taiwan. Dr. Liao is a senior member of IEEE, and a member of ACM. His research interests include artificial intelligence, image processing, pattern recognition, data mining, e-learning, management information systems, enterprise resource planning, e-commerce, smart manufacturing, and software engineering.

Cheng-Ying Yang received the M.S. degree in Electronic Engineering from Monmouth University, New Jersey, in 1991, and Ph.D. degree from the University of Toledo, Ohio, in 1999. He is a member of IEEE Satellite & Space Communication Society. Currently, he is employed as a Professor at Department of Computer Science, University of Taipei, Taiwan. His research interests are performance analysis of digital communication systems, error control coding, signal processing and computer security.

Min-Shiang Hwang received M.S. in industrial engineering from National Tsing Hua University, Taiwan in 1988, and a Ph.D. degree in computer and information science from National Chiao Tung University, Taiwan, in 1995. He was a distinguished professor and Chairman of the Department of Management Information Systems, NCHU, during 2003-2011. He obtained 1997, 1998, 1999, 2000, and 2001 Excellent Research Awards from the National Science Council (Taiwan). Dr. Hwang was a dean of the College of Computer Science, Asia University (AU), Taichung, Taiwan. He is currently a chair professor with the Department of Computer Science and Information Engineering, AU. His current research interests include information security, electronic commerce, database, data security, cryptography, image compression, and mobile computing. Dr. Hwang has published over 300+ articles on the above research fields in international journals.

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.