IJNS
International Journal of Network Security

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

Vol. 21, No. 6 (Nov. 2019)

Volume: 21, No: 6 (November 1, 2019)

International Journal of Network Security

Joint Source-Relay Selection Scheme for Cooperative Networks under Eavesdropping Environment

Jianbin Xue[1], Heng Zhu[1], Xiaoming Liao[1], and Zhe Su[2]
(Corresponding author: Heng Zhu)
School of Computer and Communication, Lanzhou University of Technology[1]
No. 287, Langongping Road, Qilihe District, Lanzhou 730050, China
School of Electronic and Information Engineering, Nanjing University of Aeronautics and Astronautics[2]
No. 29, Jiangjun Avenue, Jiang Ning District, Nanjing 210096, China
(Email: 490853309@qq.com)
(Received Mar. 27, 2018; Revised and Accepted Aug. 18, 2018; First Online June 24, 2019)

Abstract

In this paper, we propose a low complexity opportunistic source-relay selection algorithm based on power allocation for multi-source multi-relay cooperative network. We consider that the channel state information of both the main channels and the wiretap channels to minimize the system secrecy outage probability. First of all, the optimal power allocation factor for each link is given when the total transmission power is limited. Secondly, the optimal selection algorithm is designed based on power allocation and the approximate expression of the secrecy outage probability is derived in the high signal-to-noise ratio regime. Numerical experimental results demonstrate that the system secrecy outage probability of the proposed algorithm is related to the number of relay nodes but has no relation with the number of relay nodes. The proposed a low complexity opportunistic source-relay selection algorithm in this paper not only reduces the computational complexity but also has lower secrecy outage probability compared with the traditional source-relay selection algorithm.

Keywords: Cooperative Communication; Decode-and-Forward; Physical Layer Security; Secrecy Outage Probability; Source-Relay Selection

1 Introduction

With the explosive development of modern digital wireless communication technology and the continual emergence of new networks, the complexity of the wireless network security [5] problem has increasingly become the focus that people concern. Meanwhile, the open channel environment of the wireless network and the mobility of the terminal equipment also bring serious potential dangers to the information security, which makes more vulnerable to malicious attacks than the wired network. How to ensure the secure transmission of confidential information in wireless network is facing a severe test, and the security of the user information is confronting with new serious challenges.

Traditional encryption methods with secret key cryptography can obtain better safely performance, nevertheless, the encryption algorithm is too complex frequently, which need at the expense of great computation cost and increase signaling overheads undoubtedly [7]. Unlike the traditional cryptographic system, physical layer security is based on Shannon theory using the uncertain characteristics of noise channel, which does not require sophisticated algorithm to present being eavesdropped and it can theoretically analyze the degree of the information leakage easily. In recent years, scholars have paid more and more attention to the physical layer security technology [3, 6]. In [16], the authors first proposed the concept of the physical layer security and pointed out that it can also ensure data transmission security when the channel condition of legitimate users is better than the wiretap channel without need to create the keys. The authors in the literature [15] investigated the physical layer security of the multiple access Gauss wiretap channel.

It is known that there is a close connection between the secrecy transmission rate and the channel quality under eavesdropping environment, the secrecy transmission rate will be extremely low even may be to zero when the quality condition of the main channel and the eavesdropper channel is approximately equal. Cooperative relay technology has been recognized as an effective technique to achieve space diversity gain by using a virtual antenna array, which can combat channel fading effectively and increase the reliability and information rate of the transmission system. On the other hand, cooperative com-

munication technology can improve the network coverage and have been adopted in industry standard, *e.g.*, the IEEE802.16j [9] standard for relay-based wireless access networks.

Recently, a lot of works in the literature [2, 8, 11, 14] focused on improving the transmission reliability of wireless communication network by multi-user diversity and multi-relay diversity. The authors of literature [14] studied the network security by combining cooperative diversity in multi-users multi-relay cooperation network scenario. In the literature [2], three opportunistic relay selection algorithms were proposed with privacy constraints in cooperative network. Cooperative communication technology is widely used, and it can be applied to mobile communication in conjunction with D2D technology, and can also be applied to ad hoc networks [1, 12] and wireless Mesh networks. Although cooperative diversity gain has the ability to improve the reliability of wireless communication network, but with the number of users and cooperative relay nodes increases, it is also more vulnerable to illegal attack because of the same information to be sent two times or more. To solve this problem, a new multi-user multi-relay scheme based on cooperative jamming was proposed aim at maximizing the secrecy transmission rate in literature [10].

For the scenario of multiple emission sources or multiple relays, the choice of source node and relay node has a decisive influence on the system performance, so it is extremely important to select the optimal source node and relay node from plenty of potential nodes. The authors of [13] proposed the relay selection strategy by adopting decode-and-forward (DF) protocol based on the minimum secrecy outage probability (SOP) under the eavesdropping environment, which has lower outage probability compare with direct communication. In [17], the physical layer security of a multi-user system was studied in cognitive radio network. In [18], the optimal relay selections scheme for DF and amplify-and-forward (AF) relay protocols were given respective and the outage probability was analyzed. The authors of [4] discussed the performance of the optimal source-relay selection.

However, all of the above researches were assumed that the secrecy transmission rate was the fixed value zero, which lack generality. In this paper, we investigate the network physical layer security of the multi-source multi-relay cooperation network under the eavesdropping environment, and propose a source-relay selection algorithm based on power allocation aim at minimizing the system SOP. Firstly, we obtain the optimal power allocation factor for any link when the total power of the source node and relay node is limited. Basis of this, we derive the optimal source-relay selection algorithm. Subsequently, we evaluate the closed-form expression of the system SOP. Simulation results show that the proposed source-relay selection algorithm has better security performance than the traditional selection algorithm.

The remainder of this paper is organized as follows. In Section 2, we give the system model of multi-source multi-relay with an eavesdropper. In Section 3, we present the traditional source-relay selection algorithms and propose the optimal selection algorithm base on power allocation. Subsequently, we give the SOP of three selection algorithms. Finally, the related conclusion is drawn in Section 4.

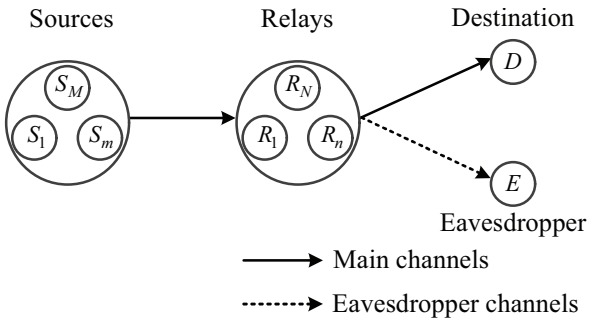


Figure 1: System model of multi-source multi-relay cooperative networks with an eavesdropper

2 System Model

As shown in Figure 1, we consider a multi-source multi-relay cooperation network, which consists of M source nodes $S_m(m = 1, 2 \cdots M)$, N candidate relay nodes $R_n(n = 1, 2 \cdots N)$, a destination node D and an eavesdropper E. In the actual communication, it is difficult for relays to transmit and receive message at the same frequency band simultaneously because of the impact of relay radio frequency devices. Therefore, all the relay nodes in this paper are equipped with a single antenna and adopt half-duplex communication mode. That is to say, the transmitted signal and the received signal must be conducted at the different time-slot.

It is assumed that there is no direct link from the source nodes to the destination node and eavesdropper, all communications must be assisted for relay and all the links follow independent flat Rayleigh fading distribution. The whole communication process is divided into two phases. In the first phase, in order to prevent the wiretap as much as possible, the optimal source-relay pair (S_{m^*}, R_{n^*}) is selected to maximum the system secrecy transmission rate. Then the selected source node S_{m^*} sends the message to the relay node R_{n^*} during this phase. In the second phase, the source sends any information no longer. The relay node R_{n^*} forwards the information received in the first phase to the destination node D as well as the eavesdropper E may overhear the information simultaneously.

In the first phase, the source S_{m^*} broadcasts the signal x_0 and the relay R_{n^*} receives the signal as:

$$y_{S_m R_n} = \sqrt{P_s} h_{S_m R_n} x_0 + n_{S_m R_n}$$

Where P_s represents the transmission power of the source S_m; $h_{S_m R_n}$ represents the channel coefficient of the link

between the source S_m and the relay R_n,which is modeled as a zero-mean complex Gaussian fading distributed with variances $\sigma^2_{R_n}$; $n_{S_mR_n}$ represents the additive Gauss white noise (AGWN) at R_n with zero-mean and variance $\sigma^2_{R_n}$.Therefore, the instantaneous signal-to-noise ratio (SNR) at R_n can be expressed as:

$$\gamma_{S_mR_n} = \frac{P_s|h_{S_mR_n}|^2}{\sigma^2_{R_n}} \tag{1}$$

In the second phase, it is assumed that the eavesdropper E knows the instantaneous channel state information (CSI) of all the relay nodes. The selected optimal relay using DF protocol to transmit data, thus the receive information at D and E can be written respectively as:

$$\begin{aligned} y_D &= \sqrt{P_i}h_{R_nD}\hat{x} + n_{R_nD} \\ y_E &= \sqrt{P_i}h_{R_nE}\hat{x} + n_{R_nE}. \end{aligned}$$

Where $\hat{x}$ represents the re-encode data symbol of the relay R_n; P_i is the transmission power of the relay R_n; h_{R_nD} and h_{R_nE} represent the channel coefficients of the links from to D and E respectively, which are modeled as the zero-mean complex Gaussian distribution with variances Ω_{R_nD} and Ω_{R_nE}; n_{R_nD} and n_{R_nE} represents the AWGN at R_n with zero-mean and variances $\sigma^2_{R_n}$ and σ^2_E, respectively. Therefore, the instantaneous SNR at D and E are expressed respectively as:

$$\gamma_{R_nD} = \frac{P_i|h_{R_nD}|^2}{\sigma^2_D} \tag{2}$$

$$\gamma_{R_nE} = \frac{P_i|h_{R_nE}|^2}{\sigma^2_E}$$

For the calculation convenience, in this paper, we let $\sigma^2_{R_n} = \sigma^2_D = \sigma^2_E = N_0$. In the cooperative communication network under eavesdropping environment, the channel $S_m \to R_n$ and $R_n \to D$ are called the main channels, and the channel $R_n \to E$ is called the eavesdropper channels. When the relay node is used DF protocol to transmit data, the relay has a certain probability of decoding errors from the source node information. Combining Equations (1) and (2), the SNR of the main channel can be expressed as $\gamma_{S_mR_nD} = \min(\gamma_{S_mR_n}, \gamma_{R_nD})$, so the information transmission rate at the node D and E can be written as:

$$\begin{aligned} C_{S_mR_nD} &= \frac{1}{2}\log_2\left[1 + \min(\gamma_{S_mR_n}, \gamma_{R_nD})\right] \\ C_{R_nE} &= \frac{1}{2}\log_2(1 + \gamma_{R_nE}) \end{aligned} \tag{3}$$

It is known that the system secrecy transmission rate is defined as the information rate difference between the main channel and the eavesdropper channel. So the system secrecy transmission rate can be expressed as:

$$C_{\text{sec}} = \frac{1}{2}\log_2\left[\frac{1 + \min(\gamma_{S_mR_n}, \gamma_{R_nD})}{1 + \gamma_{R_nE}}\right]^+ \tag{4}$$

Where $[x]^+ = \max(0, x)$, thus the system SOP can be defined as:

$$P^{\text{out}}_{\text{sec}} = \Pr(C_{\text{sec}} < R). \tag{5}$$

Where R indicates the system secrecy transmission rate threshold.

3 Joint Source-Relay Selection and Performance Analysis

In the process of source-relay pair selection, for each communication, the choice of the optimal source-relay should satisfy the maximum system secrecy transmission rate from Equation (3). In this section, we mainly explore the selection algorithm. Firstly, we give the average SOP of the random source-relay selection algorithm and the traditional source-relay selection algorithm. Secondly, we and derive the closed form expression of the system SOP. It's assumed that the total power of all nodes is P_T, let $P_T/N_0 = \gamma$. The transmission power between source node and relay is equal when using the random selection algorithm and the traditional selection algorithm, that is $P_s = P_i = P_T/2$.

3.1 Random Source-Relay Selection algorithm

There are $M \cdot N$ combinations of the random source-relay selection algorithm, one of which can be chosen with equal probability $1/(M \cdot N)$. When (S_m, R_n) is selected for communication, combining Equations (4) and (5), at the high SNR regime, the system SOP can be derived as:

$$\begin{aligned} P^{\text{out}}_{\text{sec},S_mR_n} &= \Pr\{\min(h_{S_mR_n}, h_{R_nD}) < g(R)h_{R_nE}\} \\ &= \int_0^\infty \{1 - \exp(-\frac{g(R)x}{\Omega_{S_mR_n}} - \frac{g(R)x}{\Omega_{R_nD}})\} f_{h_{R_nE}}(x)dx \\ &= 1 - \frac{1}{\Omega_{R_nE}}(\frac{g(R)}{\Omega_{S_mR_n}} + \frac{g(R)}{\Omega_{R_nD}} + \frac{1}{\Omega_{R_nE}})^{-1} \end{aligned}$$

Where $g(R) = 2^{2R}$, when adopt the random selection algorithm, the system average SOP can be derived as:

$$\begin{aligned} P^{\text{out}}_{\text{sec,ave}} &= \frac{1}{M \cdot N}\sum_{m=1}^{M}\sum_{n=1}^{N} P^{\text{out}}_{\text{sec},S_mR_n} \\ &= \frac{1}{M \cdot N}\sum_{m=1}^{M}\sum_{n=1}^{N}\{1 \\ &\quad - \frac{1}{\Omega_{R_nE}}\left(\frac{g(R)}{\Omega_{S_mR_n}} + \frac{g(R)}{\Omega_{R_nD}} + \frac{1}{\Omega_{R_nE}}\right)^{-1}\} \end{aligned} \tag{6}$$

3.2 Traditional Source-Relay Selection algorithm

Similarly to the random selection algorithm, the traditional source-relay selection algorithm is chosen under the equal power transmission of each node. In order to reduce

the probability of eavesdropping in the transmission process, the traditional scheme selects the best source-relay pair for each communication to meet the maximum secrecy transmission rate, so (S_{m^*}, R_{n^*}) can be expressed as:

$$(S_{m^*}, R_{n^*}) = \arg\max \frac{1+\min(\gamma_{S_m R_n}, \gamma_{R_n D})}{1+\gamma_{R_n E}} \quad (7)$$

Combining Equations (3) and (5), when (S_{m^*}, R_{n^*}) is selected to participant in communication and the system SOP can be expressed as:

$$\begin{aligned} P_{\text{sec}}^{\text{out}} &= \Pr\left[\max_n \frac{\min(h_{S_{m^*}R_n}, h_{R_n D})}{h_{R_n E}} < g(R)\right] \\ &= \prod_{n=1}^{N} \underbrace{\Pr\left[\min(\max_m h_{S_m R_n}, h_{R_n D}) < g(R) h_{R_n E}\right]}_{\Phi} \end{aligned}$$

Let $U = \min(\max_m h_{S_m R_n}, h_{R_n D})$, the CDF of U can be derived as:

$$\begin{aligned} &\Pr(U<u) = \Pr\left[\min(\max_m h_{S_m R_n}, h_{R_n D}) < u\right] \\ &= 1 - \Pr(h_{R_n D} > u)\Pr\left(\max_m h_{S_m R_n} > u\right) \\ &= 1 - \exp\left(-\frac{u}{\Omega_{R_n D}}\right) \\ &\quad + \exp\left(-\frac{u}{\Omega_{R_n D}}\right) \prod_{m=1}^{M}\left[1-\exp\left(-\frac{u}{\Omega_{S_m R_n}}\right)\right] \end{aligned}$$

By polynomial theory, $\prod_{m=1}^{M}\left[1-\exp\left(-\frac{u}{\Omega_{S_m R_n}}\right)\right]$ can be expanded as:

$$\begin{aligned} &\prod_{m=1}^{M}\left[1-\exp\left(-\frac{u}{\Omega_{S_m R_n}}\right)\right] \\ &\quad = 1 + \sum_{m=1}^{2^M-1} (-1)^{|S_j|} \exp(-\sum_{m\in S_j} \frac{u}{\Omega_{S_m R_n}}). \end{aligned}$$

Where S_j represents the j-th non-empty collection, $|S_j|$ denotes the cardinality of set S_j. Substituting Equation (7) into (6), we can have Equation (8):

$$\begin{aligned} \Phi &= \int_0^{\infty} [1-\exp\left(-\frac{g(R)u}{\Omega_{R_n D}}\right) + \exp\left(-\frac{g(R)u}{\Omega_{R_n D}}\right) \\ &\quad \cdot \left(1 + \sum_{j=1}^{2^M-1} (-1)^{|S_j|} \exp(-\sum_{m\in S_j} \frac{g(R)u}{\Omega_{S_m R_n}}\right)] \\ &\quad \cdot f_{\gamma_{R_n E}}(u)\, du \\ &= 1 + \frac{1}{\Omega_{R_n E}} \sum_{j=1}^{2^M-1} (-1)^{|S_j|} \\ &\quad \cdot \left(\frac{g(R)}{\Omega_{R_n D}} + \frac{g(R)}{\Omega_{R_n E}} + \sum_{m\in S_j} \frac{g(R)}{\Omega_{S_m R_n}}\right)^{-1} \end{aligned} \quad (8)$$

3.3 Optimal Source-Relay Selection algorithm

The above two selection algorithms were designed by minimizing the system SOP in the case of equal power transmission for all nodes, which was a sub-optimal selection method with high computational complexity. In this section, we propose a lower complexity source-relay selection algorithm, which based on power allocation to search the optimal source and relay. That is, we obtain the power allocation factor of each link potential participation nodes firstly and then select the optimum source-relay (S_{m^*}, R_{n^*}).

3.3.1 Power Allocation Process

When $(S_m R_n)$ is selected to participate in cooperation communication, if the transmission power of the source node S_m is $\mu P_T (0<\mu<1)$, so the transmission power of the relay node R_n is $(1-\mu)P_T$. Thus the system SOP can be expressed as:

$$C_{(m,n)} = \frac{1+\min(\gamma_{S_m R_n}, \gamma_{R_n D})}{1+\gamma_{R_n E}}$$

When $\gamma_{S_m R_n} < \gamma_{R_n D}$, $C_{(m,n)} = \frac{1+\mu\gamma|h_{S_m R_n}|^2}{1+(1-\mu)\gamma|h_{R_n E}|^2}$. Because $\frac{\partial C_{(m,n)}}{\partial \mu}$ is true forever $C_{(m,n)}$ is a strictly monotone increasing function. We can enhance μ until $\gamma_{S_m R_n} = \gamma_{R_n D}$, the function $C_{(m,n)}$ can get the maximum value now.

When $\gamma_{S_m R_n} > \gamma_{R_n D}$, $C_{(m,n)} = \frac{1+(1-\mu)\gamma|h_{R_n D}|^2}{1+(1-\mu)\gamma|h_{R_n E}|^2}$.Thus $\frac{\partial C_{(m,n)}}{\partial \mu} = A(|h_{R_n E}|^2 - |h_{R_n D}|^2)$, the variable A is a factor greater than 0 in the formula. If $|h_{R_n D}|^2 > |h_{R_n E}|^2$, $C_{(m,n)}$ is a monotonically decreasing function. We can enhance μ until $\gamma_{S_m R_n} = \gamma_{R_n D}$, the function $C_{(m,n)}$ gets the maximum value now. If $|h_{R_n D}|^2 < |h_{R_n E}|^2$, no matter how we change μ, the system secrecy transmission rate is 0 because of $C_{(m,n)} < 1$.

To summarize, if the source-relay pair (S_m, R_n) is selected to participates in cooperative communication, the optimal power allocation of each transmission can be expressed as:

$$\gamma_{S_m R_n} = \gamma_{R_n D}, \texttt{that is } \mu = \frac{|h_{R_n D}|^2}{|h_{S_m R_n}|^2 + |h_{R_n D}|^2} \quad (9)$$

3.3.2 Optimal Source-Relay Selection Process

Substituting Equation (9) into (7), the optimal source-relay pair (S_{m^*}, R_{n^*}) is given in Equation (10):

$$\begin{aligned} &(S_{m^*}, R_{n^*}) = \\ &\arg\max_{m,n}\left\{\frac{1}{2}\log_2 \frac{1+\gamma|h_{S_m R_n}|^2|h_{R_n D}|^2/(h_{S_m R_n}|^2+|h_{R_n D}|^2)}{1+\gamma|h_{S_m R_n}|^2|h_{R_n E}|^2/(h_{S_m R_n}|^2+|h_{R_n D}|^2)}\right\} \end{aligned} \quad (10)$$

Combining Equations (5) and (10), when SNR is large

enough, the system SOP can be expressed as:

$$P_{\text{sec}}^{\text{out}} = \Pr\left[\max_{n}\left(|h_{R_nD}|^2/|h_{R_nE}|^2\right) < g(R)\right] = \prod_{n=1}^{N} \underbrace{\Pr\left(|h_{R_nD}|^2 < g(R)|h_{R_nE}|^2\right)}_{\Psi} \tag{11}$$

Where Ψ can be obtained as:

$$\Psi = \int_0^{\infty} \Pr\left[|h_{R_nD}|^2 < xg(R)\right] f_{h_{R_nE}}(x)dx = \frac{g(R)\Omega_{R_nE}}{\Omega_{R_nD} + g(R)\Omega_{R_nE}} \tag{12}$$

Therefore, when the source-relay pair (S_{m^*}, R_{n^*}) is selected and the system SOP of the proposed algorithm can be expressed as

$$P_{\text{sec}}^{\text{out}} = \prod_{i=1}^{N} \frac{g(R)\Omega_{R_nE}}{\Omega_{R_nD} + g(R)\Omega_{R_nE}} \tag{13}$$

In conclusion, when the number of source nodes is M and the number of relay nodes is N, the traditional selection algorithm need to compare the performance of the $M \cdot N$ links, and also need to calculate the integral polynomial multiplication with high computational complexity.

In our proposed algorithm, the optimal allocation factor among the source node and relay node of any link is obtained by power allocation and we can obtain the system SOP easily. It can be seen from Expression (13) that the SOP of the proposed scheme is independent function of the source node. The relay nodes can decode correctly by regulating the transmission power of source node and reduce the computational complexity to a great extent.

3.4 Numerical Results and Discussions

In this section, we analyze the SOP performance of the proposed source-relay selection algorithm by Monte-Carlo simulation, and compared it with the random selection and the traditional selection of two source-relay selection algorithms. In the simulations, we simulate a line network and assume that not only M source nodes but also N relay nodes are distributed in the same position respectively. The distance between the source and the destination is normalized to one, and all the relay nodes are located at the precise middle between the source and the destination. Therefore, the channel coefficient of any link follows the complex Gauss random distribution with zero-mean and variance d_{ij}^{-v}, where d_{ij} denotes the distance between any two nodes and stands for the path-loss factor. Here, we set $v = 4$ for an urban environment. The main-to-eavesdropper ratio was defined as the ratio of the main channel gain over the eavesdropper channel gain (*i.e.*, MER=$\Omega_{S_mR_n}/\Omega_{R_nE}$). In addition, the system target secrecy information rate is supposed as $R = 1\text{bit}/(s \cdot \text{Hz})$.

Figure 2 shows the theoretical value curves and the simulation values of the system SOP for different number

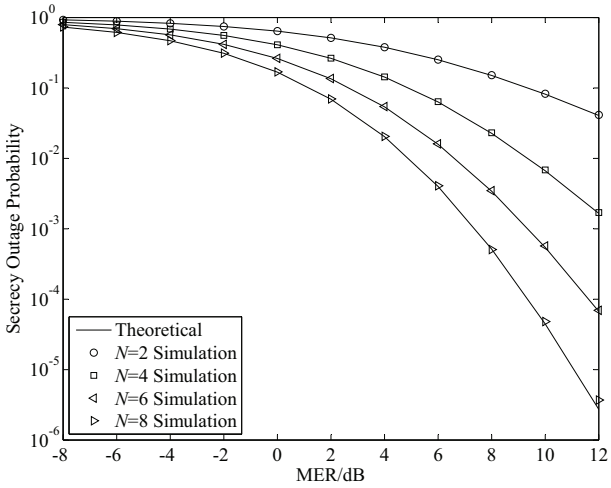


Figure 2: Theoretical values and simulation values of the secrecy outage probability with the different relays number

of candidate relays with the same source nodes number M=4. It can be seen from the figure, the theoretical value curve of the proposed algorithm are approximately coincidence with the simulation curve for the case of N=2, N=4, N=6 and N=8. Thus we prove the positive solution of the proposed algorithm. As well with the same number of nodes, the SOP of the four curves shows descend trend along with the increasing of MER. This is because the quality of the wiretap channel get worse compare with the main channel when increases, so the secrecy transmission rate is becoming larger. Therefore, in order to ensure the transmission more secure, one way is to move the eavesdroppers far away from the sources. Meanwhile, with the same channel condition, the more the number of candidate relays, the smaller the SOP.

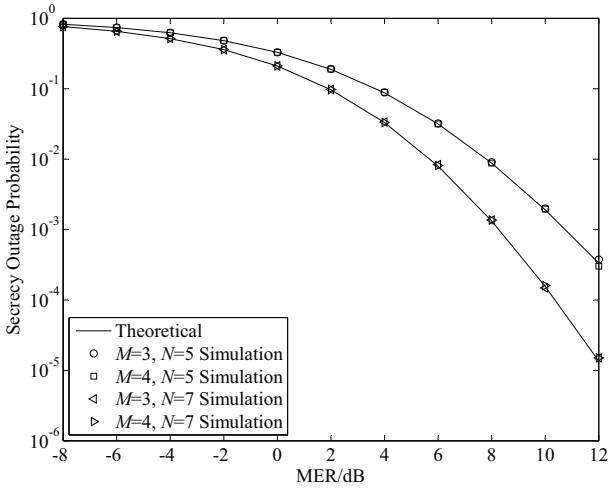


Figure 3: Comparison of the SOP under different channel conditions

Figure 3 gives the theoretical value curves and the simulation values of the system SOP when the number of source nodes and relay nodes is different. It can be seen

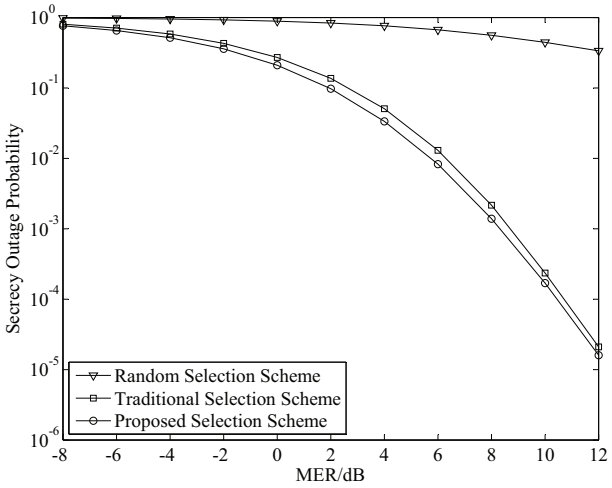


Figure 4: Comparison of secrecy outage probability between the different source-relay selection algorithms (M=4, N=5)

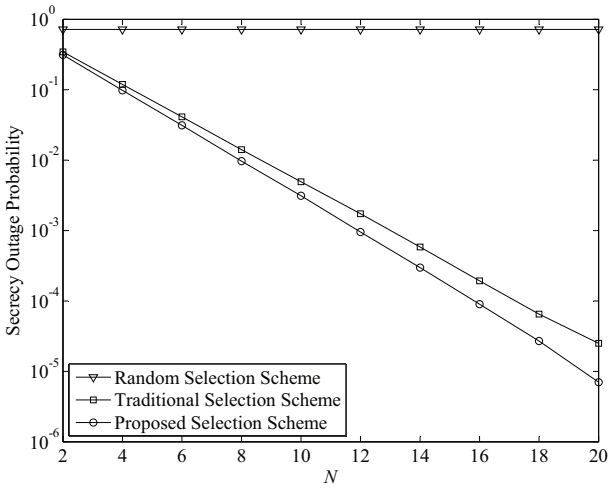


Figure 5: Comparison of secrecy outage probability between different schemes with the same candidate relay number (M=5, MER=5dB)

that the theoretical value curve of the proposed source-relay selection algorithm approximately coincides with the simulation curve when the number of candidate source nodes and relay nodes is constant. Thus the correctness of the proposed algorithm is further verified. At the same time, we can see that the system SOP decreases with the increasing of MER. When the number of relay nodes remain unchanged, the SOP of the algorithm are constant no matter how many the number of the candidate source nodes, which shows that the proposed selection algorithm related to the number of candidate relays but independent of the number of source nodes. This is because when the number of candidate relay nodes is fixed, we can adjust the power of each link in the proposed algorithm, and by calculating the system secrecy transmission rate is a function which independent of the source node channel.

Figure 4 illustrates the system SOP curves of the different source-relay selection algorithms under the same channel quality condition. We can see that the SOP of the three selection algorithms turn out descend trend with the increasing of MER. In the same channel condition, the random selection algorithm has the highest outage probability. However, the traditional selection algorithm chooses the source-relay with the highest security transfer rate to participate in cooperative communication under equal power allocation and the system SOP is significantly reduced compared with the random selection algorithm. The proposed scheme in this paper has the lowest outage probability, the optimal power allocation is firstly carried out for each link, and the source-relay pair is selected according to the quality of the main channel and the wiretap channel subsequently.

Figure 5 presents the system SOP curves of the different source-relay selection algorithms under the same candidate relays condition. We can see from the figure, when the number of source nodes is fixed, the SOP of the random selection algorithm is independent of the number of candidate relay nodes because the source nodes and relay nodes are distributed in the same position, which is linear. Meanwhile, the system SOP of the traditional scheme and the proposed scheme turn out descend trend with the increasing of candidate relay node N. In the same number of resource nodes and relay nodes, the random source-relay selection algorithm has the highest SOP, and the traditional selection algorithm is secondary. The SOP of the proposed scheme in this paper is the lowest.

4 Conclusions

In this paper, a new opportunistic source-relay selection algorithm is proposed for the multi-source multi-relay cooperative networks, which aims at minimizing the system SOP. We joint considering the CSI of the main channel and the wiretap channel. Firstly, the optimal power allocation factor for any link is obtained when the total transmitted power is limited, which is a function of the channel statistics .On the basis of this, the selection algorithm of the optimal source-relay is given and the closed-form expression of the system SOP is derived. The simulation results verify the proposed scheme has lower SOP performance compared with the traditional selection algorithm and reduce the computational complexity.

In addition, this work is assumed that there is no direct link between all source nodes and destination node. In the future work, we will continue to explore the physical layer security for the multi-source multi-relay networks with direct link.

Acknowledgments

This study was supported in part by National Natural Science Foundation of China (NO. 61461026), and Open Research Fund of National Mobile Communications Research Laboratory, Southeast University (N0.

2014D13).The authors would like to thank the anonymous reviewers and the editors for their suggestions.

References

[1] T. Alam and M. Aljohani, "Design a new middleware for communication in Ad Hoc network of Android smart devices," in *International Conference on Information and Communication Technology for Competitive Strategies*, Dec. 2016. (https://www.researchgate.net/publication/307080049_Design_a_New_Middleware_for_Communication_in_Ad_Hoc_Network_of_Android_Smart_Devices)

[2] V. N. Q. Bao, N. Linh-Trung, and M. Debbah, "Relay selection algorithms for dual-hop networks under security constraints with multiple eavesdroppers," *IEEE Transactions on Wireless Communications*, vol. 12, no. 12, pp. 6076-6085, 2013.

[3] M. Bloch and J. Barros, *Physical-Layer Security: From Information Theory to Security Engineering (1st ed)*, 2011. (https://www.cambridge.org/core/books/physicallayer-security/543CF3D1431805B6AE04A7AA72903D09)

[4] W. F. Cao, Y. L. Zou, and Z. Yang, "Joint source-relay selection for improving wireless physical-layer security," in *Proceedings of The 59th IEEE Global Communications Conference (GLOBECOM'16)*, Dec. 2016. (https://ieeexplore.ieee.org/abstract/document/7841935)

[5] G. J. Chen, Y. Gong, P. Xiao, and J. A. Chambers, "Physical layer network security in the Full-Duplex relay system," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 3, pp. 574-583, 2015.

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

[7] K. Emura, G. Hanaoka, Y. Sakai, and J. C. N. Schuldt, "Group signature implies public-key encryption with non-interactive opening," *International Journal of Information Security*, vol. 13, no. 1, pp. 51-62, 2014.

[8] L. S. Fan, X. F. Lei, T. Q. Duong, E. Maged, and G. K. Karagiannidis, "Secure multiuser communications in multiple amplify-and-forward relay networks," *IEEE Transactions on Communications*, vol. 62, no. 9, pp. 3299-3310, 2014.

[9] V. Genc, S. Murphy, Y. Yu, and J. Murphy, "IEEE 802.16J relay-based wireless access networks: An overview," *IEEE Wireless Communications*, vol. 15, no. 5, pp. 56-63, 2008.

[10] S. I. Kim, I. M. Kim, and J. Heo, "Secure transmission for multiuser relay networks," *IEEE Transactions on Wireless Communications*, vol. 14, no. 7, pp. 3724-3737, 2015.

[11] X. F. Lei, L. S. Fan, R. Q. Hu, D. S. Michalopoulos, and P. Z. Fan, "Secure multiuser communications in multiple decode-and-forward relay networks with direct links," in *IEEE Global Communications Conference (GLOBECOM'14)*, pp. 3180-3185, Dec. 2014.

[12] V. S. Naresh, and N. V. E. S. Murthy, "Elliptic curve based dynamic contributory group key agreement protocol for secure group communication over Ad-hoc networks," *International Journal of Network Security*, vol. 17, no. 5, pp. 588-596, 2015.

[13] P. N. Son and H. Y. Kong, "Exact outage probability of a decode-and-forward scheme with best relay selection under physical layer security," *Wireless Personal Communications*, vol. 4, no. 2, pp. 325-342, 2014.

[14] L. Sun, T. Y. Zhang, L. Lu, and H. Niu, "On the combination of cooperative diversity and multiuser diversity in multi-source multi-relay wireless networks," *IEEE Signal Processing Letters*, vol. 17, no. 6, pp. 535-538, 2010.

[15] E. Tekin and A. Yener, "The Gaussian multiple access wire-tap channel: Wireless secrecy and cooperative jamming," in *Information Theory and Applications Workshop (ITA'07)*, pp. 404-413, Feb. 2007.

[16] A. D. Wyner, "The wire-tap channel," *Bell System Technical Journal*, vol. 54, no. 8, pp. 1355-1387, 1975.

[17] Y. L. Zou, X. B. Wang, and W. M. Shen, "Physical-layer security with multiuser scheduling in cognitive radio networks," *IEEE Transactions on Communications*, vol. 61, no. 12, pp. 5103-5113, 2013.

[18] Y. L. Zou, X. B. Wang, and W. M. Shen, "Optimal relay selection for physical-layer security in cooperative wireless networks," *IEEE Journal on Selected Areas in Communications*, vol. 31, no.10, pp. 2099-2111, 2013.

Biography

Jianbin Xue is a professor and deputy dean of school of computer and communication, Lanzhou University of Technology, China. He received the B.S. degree in communication engineering from Sichuan University, in 1997. He received the M.S. degree and the Ph.D. degree both from Lanzhou University of Technology, Lanzhou, China, in 2005 and 2009, respectively. His main research interests include wireless communication theory and technology, wireless network theory and technology.

Heng Zhu received the B.S. degree in electronic and information engineering from Shihezi University, China, in 2010. He is currently pursuing his M.S. degree in the school of Computer and Communication, Lanzhou University of Technology. His research interests include cooperative communication and D2D communication.

Xiaoming Liao received the B.S. degree from Wuhan University of Science and Technology, in 2014. He is currently pursuing his M.S. degree in the school of Computer and Communication, Lanzhou University of Technology.

His main research interests is wireless heterogeneous network.

Zhe Su was born in Inner Mongolia Province, China, in 1993. He is currently pursuing his Ph.D. degree in the school of Electronic and Information Engineering, Nanjing University of Aeronautics and Astronautics. His major is Communication and Information System.

An Anti-counterfeit Complete RFID Tag Grouping Proof Generation Protocol

Gao-Feng Shen[1], Shu-Min Gu[2], and Dao-Wei Liu[3]
(Corresponding author: Gao-Feng Shen)
Department of Computer Science, Zhengzhou University of Light Industry[1]
Zhengzhou 450002, China
(Email: shgf_123@163.com)
Department of Basic Course, College of Information and Business, Zhongyuan University of Technology[2]
Department of Computer Science and Engineering, Guangzhou College of Technology and Business[3]
(Received May 27, 2018; Revised and Accepted Oct. 18, 2018; First Online June 13, 2019)

Abstract

Most existing radio frequency identification group tags prove that the generation protocol does not meet the lightweight Gen-2 standard and there are existing security issues such as rapid brute-force cracking, proof forgery, and incomplete authentication. To tackle with these problems, an improved anti-counterfeit complete RFID tag grouping proof generation protocol was designed. The protocol adopted a one-way pseudo-random function that conformed to the low-cost Gen-2 standard as the basic encryption algorithm to implement complete tripartite authentication, label group proof generation and verification. The encryption authorization Mark identification was used by the database and the cipher-text transmission mechanism was ures to avoid multiple malicious attacks on the protocol. Finally, the feasibility of the protocol was approved by the GNY logic, and the security attack description indicated that the improved protocol met the security standards. Performance comparison analysis showed that the protocol was in line with low-cost applications.

Keywords: Cipher-text Transmission; Mark Identity; Proof Generation; RFID; The Gen-2 Standard

1 Introduction

Radio frequency identification (RFID) is a technology for sensing and recognizing a designated object through radio signals to process related data. With the continuous application and development of Internet of Things technology, RFID technology has been known and applied widely [9]. There into, the RFID tag grouping proof mechanism, that is, the proof mechanism that provides a group of RFID tags coexisting at a certain time, is getting more and more attention. This mechanism is playing a very important role in ensuring the security and integrity of the tag's entity and can be applied to a variety of scenarios. For example, in the field of medicine, it should be ensured that patients' multiple medications are delivered at exactly the same time, or a certain drug and its instructions for usage must be enclosed at the same time as it was sold. Similarly, in the field of equipment production, manufacturers need to assure the purchasing organization that all parts and components of a certain equipment are delivered at the same time and that the parts have indeed left the factory along with their safety shields [5,11,13,17]. In these scenarios, it is not adequate just to ensure the security of a single-tag entity, but to uniformly verify the entity of the multiple tags in a group. Therefore, designing a RFID tag grouping proof protocol with high security and suitability for low-cost tag applications is currently one of the hottest research topics. Reference [9] proposed the concept of conjugate proof of RFID tags for the first time, and the simultaneous existence proof showed that the system could generate two tags at the same time. Reference [9] further gave the corresponding proof for the generation protocol. With the introduction of the concept of label conjugation proof, many protocol researchers had been furtherly developed the grouping proof schemes for multiple tags within a group, in which the simultaneous existence of a group of tags were certified by the reader and group tags could be generated in a predetermined time interval.

The related grouping proof documents are as follows: Reference [1] proposed an ultra-lightweight tag grouping proof protocol based on simple bit operations. The protocol was simple and the cost was low for large-scale production applications. But it was also due to this simplicity and lightweight of the protocols, they were vulnerable to forgery and counterfeit attacks. Later, Reference [8] proposed a more secure tag grouping proof protocol based on heavyweight public key cryptography. However, when the protocol was executed, the calculation process was complex, and the tag computation cost was high, which was

not suitable for practical applications of low-cost passive tags. Reference [18] proposed a tag grouping proof protocol based on general combination security [19]. This protocol did not require a trusted third party, but only needed a pseudo-random number generator which would perform bit manipulation, and to a certain extent, could reduce the system cost. However, being analyzed in [10], the protocol was vulnerable to message integrity attacks. Reference [2] proposed an improved RFID tag grouping proof protocol based on the message verification code function, but the article failed to demonstrate the theory explicitly, and its security still needs to be systematically proved. In the follow-up, researchers proposed a more secure symmetric cryptographic tag grouping proof protocol based on hash function and one-way pseudo-random function to meet the requirement of application of lightweight cost [3, 6]. Reference [6] proposed a tag grouping proof protocol based on the hash function. There was no dependency between the tags and they had high reliability. However, it was found in this study that the tag grouping before the reader authentication processes the generation was proved to be slightly duplicated and not concise; and the agreement could not achieve full authentication where there existed the threat of counterfeit attacks. Reference [3] was based on a one-way pseudo-random function and proposed a lightweight group privacy protection protocol. However, through the research of this paper, it was found that the Reference [3] showed inadequate resistance to brute force attack and proof forgery threat.

The rest of the paper was organized as follows: The second part analyzed the security vulnerabilities in references [6] and [3]. The third part proposed its own improved protocol for the security loopholes in the above references [6] and [3]. The fourth part gave the proof of the formalization of GNY logic of the improved protocol, which demonstrated feasibility and legitimacy of this protocol. The fifth part performed a security analysis of the protocol. It also compared the related documents. A conclusion was drawn that this protocol had higher security. The sixth part gave a comparison of the performance of the protocols and related literature, which showed that the agreement costed less and had high efficiency. The seventh part gave a summary and outlook.

2 References [6] and [3] Security Vulnerability Analysis

The security analysis of RFID tag grouping proof protocol based on hash algorithm proposed by Reference [6] was as follows:

Because Reference [6] did not have a complete safety certification process, and only had the mutual authentication process between the reader and the tag, it lacked verifier's verification process for the reader and tag. Therefore, the attacker could directly generate the random number R_1 and R_2 and even the information $M = MAC[m_1, m_2, m_3 \ldots]$ through the counterfeit reader. The computation factor m_i in M could be obtained by an attacker eavesdropping on c_t using the random number R_1 generated by impersonation $m_i = MAC[c_t, R_{i1}]$. Finally, because the protocol lacked the verifier's verification process for the reader, the attacker would forge the grouping proof P, $P = (M, R_1, R_2, \ldots)$, which the verifier would still automatically verify. Therefore, Reference [6] presented the security vulnerability of incomplete authentication and the loophole for forgery attack.

The security analysis of RFID tag grouping proof protocol based on one-way pseudo-random function proposed by Reference [3] was as follows:

Reference [3] had the threat from brute force because the security of the Reference [3] depended entirely on the choice of encryption function, and the secret information N_1-N_5 was transmitted in plain text. In the communication data $r_i = g(PID_i \oplus N_2)$ and $r_R = g(PID_i \oplus N_3)$, r_i, r_R,$N2$-$N3$, the pseudo-random algorithm $g()$ was disclosed. Only the PID_i was unknown, and the attacker could easily use illegal interception and interception to perform a brute force attack and obtain the tag identification information PID_i, so that the tags were maliciously tracked and the brute-force attacks succeed.

Reference [3] had proved flawed because the attacker could quickly decrypt the tag key information S_i according to the public hash algorithm, communication data m, $m_i = h(S_i, r_i')$ and the eavesdropped plain text data r_i'. Meanwhile, the important component of the protocol identification P was the identifier ticket. Because S_i and PID_i had been obtained, according to the formula $V = E_{K_i}(ticket, PID_i)$, the attacker could crack the session secret identifier ticket, and then falsify tag grouping proof P, $P = h(K||ticket||m_1 \oplus \ldots \oplus m_n)$. The forgery attack was thus successful.

In summary, this article aimed to improve the functions of above-mentioned references [3, 6], including the resistance against brute-force attacks, forgery attacks, the loophole of incomplete authentication, and the defect of high system cost and complexity, and to forge a more secure, fully-certified RFID tag grouping proof generation protocol that met the Gen-2 standard. In order to reduce system cost, this protocol used a pseudo-random function and did not use the hash function because the implementation of the Gen-2 standard in the Global Product Electronic Code Center had become the design standard for the RFID tag industry. The Gen-2 standard stipulated that only 2500-5000 circuits in the tag could be used for calculation. The commonly used hash function (MD5 requires 15,000-20,000 circuits) was not suitable for the Gen-2 standard [12,14]. Pseudo-random functions and some simple bit operations for designing protocols for security standards were attracting more and more attention.

3 Improved Tag Grouping Proof Protocol

3.1 Prerequisites and Symbols

This agreement did not require trusted third party to support only tags, readers, and databases. The database was usually a trusted entity that was physically secured and difficult to be invaded in RFID system applications. It often stored secret information (eg: keys, identities, *etc.*) needed by the system. The reader wired connection to which the database it was connected could be regarded as a secure communication channel. Assuming that the keys stored in the tag were difficult to steal, side channel attacks [15] and physical cloning attacks [7] were beyond the scope of this article. And for the wireless connection between the reader and the tag: it was precisely due to the openness of this communication, there were a large range of attack behaviors applicable between them, which could be regarded as an insecure communication channel. The basic characteristics of the communication with the general tag grouping proof protocol were the same. It was assumed that this protocol would be completed within a predetermined time interval. The symbols appeared in the agreement were listed in Table 1.

Table 1: Symbol description

Symbol	Meaning
T	Tag
R	Reader
D	Database
PID_T	Tag pseudonym ID
PID_R	Reader pseudonym ID
K_i	Tag key
K_R	Reader key
K_g	Group tags Shared Key
N_1, N_2, N_3	Removed
$A-G$, P, m_i	Three-way communication data
$Mark$	Authorization mark
$g_k(x, y)$	Pseudo-random function based on shared key

3.2 Agreement Specific Certification Process

This agreement was divided into five stages, i.e. initialization, authorization, mutual authentication, grouping proof generation and authentication, and key update. Figure 1 shows the specific implementation flow of the improved fully-certified RFID tag grouping proof generation protocol, whose process is explained as follows:

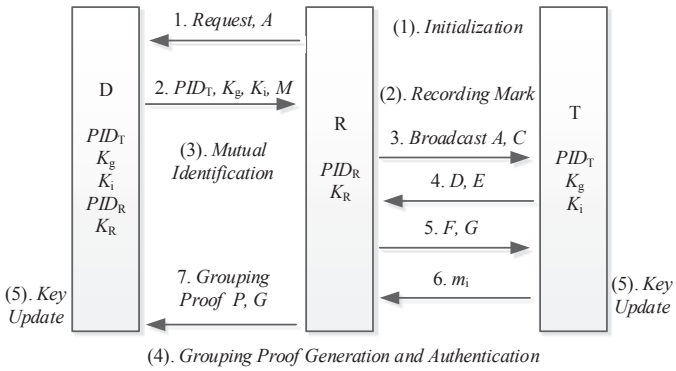


Figure 1: Improved RFID tag grouping proof generation protocol

3.2.1 Initialization

The three parties shared a one-way pseudo-random function with low complexity. The single tag in each tag group in the database corresponded to $\{PID_T, K_i, K_g\}$. The corresponding information of the reader was recorded as $\{PID_R, K_R\}$. The group single label stored its own pseudonym identifier PID_T and key information K_i and group shared key K_g; the reader/writer stored its own pseudonym identifier PID_R and key information K_R.

3.2.2 Authorization

Database pre-authenticated readers, generating authorization identifiers $Mark$, in preparation for subsequent group certification generation. The specific process was as follows:

Step 1. The reader first sent a group tag authentication authorization requesting a pre-authentication message A from the database.

$$A = K_{Ri} \oplus PID_{Ri}.$$

Step 2. After the database received the request instruction, it calculated and tried to find whether there was a reader record equal to A according to all the reader information stored in it. If it did not exist, it meant the reader might be impersonated and the authorization would be terminated. If it existed, it meant that the reader was legitimate, and the database would send pre-authenticated group tag information $\{(PID_{Ti}, K_i)|1 \leq i \leq n, K_g\}$ and message M to the reader. The message M was composed of the authorization identifier Mark generated by the database random number generator.

$$M = \mathrm{g}_{Kg}(K_g) \oplus Mark.$$

Step 3. The reader sequentially stored the group tag information transmitted from the database, and decrypted the authorization flag $Mark$ using the received group key K_g, and the reader would obtain the authentication and authorization successfully.

3.2.3 Mutual Authentication

The authorized reader started the process of mutual identification and identification with the group tag, which was the basis for generating the grouping proof P. Assuming that one tag PID_{T_i} was selected in the group tag, the specific authentication process was as follows:

Step 1. The reader used the random number generator to generate a random number N_1 and broadcasted the encrypted messages B, C to the tag.

$$B = \mathrm{g}_{Kg}(K_g \oplus PID_{Ti}), C = K_g \oplus N_1.$$

Step 2. After the tag in the group received the message broadcasted by the reader, each tag in the group was calculated using its own key K_i and pseudonym PID_{T_i}. When the calculated message B was found to be equal to the received B, the tag was activated, and the random number N_1 ($N_1 = C \oplus K_g$) was decrypted to generate a message to act as authenticated D. Subsequently, the activated tag generated a random number N_2 and an encrypted message E, and finally D and E were sent to the reader.

$$D = \mathrm{g}_{K_{Ti}}(PID_{Ti} \oplus K_{Ti} \oplus N_1), E = K_g \oplus N_2.$$

Step 3. When the reader received the replying message from the activated tag, it would verify whether $D' = D$ or not. If they were not equal, the label was illegal and the agreement was terminated. If they were equal, the reader would authenticate the tag as a legal tag and the protocol would continue. The reader would decrypt and obtain the random number N_2 ($N_2 = E \oplus K_g$), and generate the information authenticated F, and continue to generate the random number N_3 and the encrypted message G and send F and G to the tag.

$$F = \mathrm{g}_{K_{Ti}}(PID_{Ti} \oplus K_{Ti} \oplus N_2), G = K_g \oplus N_3.$$

Step 4. After the activated tag received the reader's message, it used its own PID_T, K_i message to verify whether $F' = F$ or not. If not, the tag verification reader failed and the authentication terminated; if they were equal, the tag was successfully authenticated and the reader performed subsequent calculations. The tag decrypted the random number N_3 ($N_3 = G \oplus K_g$), updating the pseudonym identifier PID_T ($PID_{Ti} = \mathrm{g}_{K_{Ti}}(PID_{Ti} \oplus K_{Ti}) \oplus N_3$) and the key information K_i ($K_{T\mathrm{i}} = K_{Ti} \oplus N_3$), and generating the group authentication and identification factor m_i and sent it to the reader.

$$m_i = \mathrm{g}_{K_{Ti}}(PID_{Ti} \oplus K_{Ti} \oplus N_3).$$

3.3 Grouping Proof Generation and Authentication

Step 1. All tags in the group repeated the mutual identification process; activating all tags in the group, the reader would receive all tag authentication identification factors m_i within a specified time. Generating group certification message P after successful reception, the reader would finally send P to the database.

$$P = \mathrm{g}_{K_g}(K_g || Mark || m_1 \oplus m_2 \ldots \oplus m_n).$$

Step 2. After the database received the grouping proof message, it first calculated the grouping proof P to determine whether it was equal to the grouping proof value P from the reader. If they were equal, the database verified the grouping proof to be successful and proved that the group tag existed at the same time. Afterwards, by decrypting the random number N_3 ($N_3 = G \oplus K_g$), the tag pseudonym identifier PID_T and the key information K_i were updated. At this point, the protocol was completed; if they were not equal, the verification failed and the protocol terminated.

$$\begin{aligned} PID_{Ti} &= \mathrm{g}_{K_{Ti}}(PID_{Ti} \oplus K_{Ti}) \oplus N_3, \\ K_{T\mathrm{i}} &= K_{Ti} \oplus N_3. \end{aligned}$$

3.4 Key Update

This process usually refers to the updating of the secret information (pseudonym, key) in the tag and database, which had already been described in the mutual authentication and grouping proof generation and authentication phases, and would not be repeated here.

4 Improved Protocol GNY Logic Proof

GNY logic was a proof of security logic form proposed by L. Gong, R. Needham and R. Yahalom *et al.* It was the most direct and simplest method of analysis. In the field of formal verification of RFID algorithms it had a more widespread application.

Its basic idea: Firstly, the algorithm operating environment and the entities involved in communication should be initially idealized; Secondly, it was necessary to define a reasonable and safe proof target according to the algorithm application requirements. In addition, the entire algorithm process was simulated using GNY logic language rules; finally, a number of GNY axioms and rules were used to derive the correct target from the algorithm process [16]. The GNY logic had a total of nearly 50 inference rules [4]. The relevant rules used in this paper are described as following:

Freshness rules F_1: $\frac{P|\equiv \#X}{P|\equiv \#(X,Y), P|\equiv \#F(X)}$.

Message interpretation rules I_1:

$$\frac{P \lhd *(X)_K, P \in K, P|\equiv P \overset{K}{\longleftrightarrow} Q, P|\equiv \phi(X), P|\equiv \#(X,K)}{P|\equiv Q|\sim X, P|\equiv Q|\sim (X)_K, P|\equiv Q \in K}.$$

Have rules P_1: $\frac{P \lhd X}{P \ni X}$.

Identifiable rules R_6: $\frac{P \ni H(X)}{P|\equiv \phi(X)}$.

4.1 Idealized Model

Using T for the label, R for the reader, D for the database, $gk()$ denoted a one-way pseudo-random function encrypted with the key K_i, $g_{K_g}()$ denoted a one-way pseudo-random function encrypted with the key K_g, the idealized model of this agreement was described as follows:

$M_1 : D \triangleleft [K_R, PID_R]$
$M_2 : R \triangleleft *[PID_T, K_T, K_g], [Mark]$
$M_3 : T \triangleleft *[N_1], g_{K_g}(K_g \oplus PID_T)$
$M_4 : R \triangleleft *[N_2], \mathrm{g}_{K_T}(PID_T \oplus K_T \oplus N_1)$
$M_5 : T \triangleleft *[N_3], \mathrm{g}_{K_T}(PID_T \oplus K_T \oplus N_2)$
$M_6 : R \triangleleft *\mathrm{g}_{K_T}(PID_T \oplus K_T \oplus N_3)$
$M_7 : D \triangleleft *[N_3], \mathrm{g}_{K_g}(K_g || Mark || m_1 \oplus m_2 \oplus \ldots \oplus m_n)$.

4.2 Initialization Assumption

$X_1 : T \in (PID_T, K_T, K_g), g_{K_T}(), g_{K_g}()$
$X_2 : R \in (PID_R, K_R, K_g), g_{K_g}()$
$X_3 : D \in (PID_T, K_T, K_g), (PID_R, K_R), g_{K_T}(), g_{K_g}()$
$X_4 : R \in N_1, N_3$
$X_5 : T \in N_2$
$X_6 : R \mid\equiv \phi(PID_T, K_g), \phi(PID_T \oplus K_T \oplus N_2),$
$\phi(K_g || Mark || m_1 \oplus m_2 \oplus \ldots \oplus m_n)$
$X_7 : T \mid\equiv \phi(PID_T \oplus K_T \oplus N_1), \phi(PID_T \oplus K_T \oplus N_3)$.

4.3 Expected Goals

Identification and certification of D-to-R identity information:

$$D_1 : D \mid\equiv \phi(PID_R, K_R).$$

Identification and certification of R-to-T identity information:

$$D_2 : R \mid\equiv T \mid\sim \#g_{K_T}(PID_T \oplus K_T \oplus N_1).$$

Identification and certification of T-to-R identity information:

$$D_3 : T \mid\equiv R \mid\sim \#g_{K_T}(PID_T \oplus K_T \oplus N_2).$$

D authenticated the received group certification:

$$D_4 : D \mid\equiv R \mid\sim \mathrm{g}_{K_g}(K_g || Mark || m_1 \oplus m_2 \oplus \ldots \oplus m_n).$$

That was, if database D calculates P' to be the same as received P (P $=\mathrm{g}_{K_g}(K_g || Mark || m_1 \oplus m_2 \oplus \ldots \oplus m_n)$), then it was believed that tag information existed at the same time. The following used P for $(K_g || Mark || m_1 \oplus m_2 \oplus \ldots \oplus m_n)$.

4.4 Reasoning Proof

After receiving the message M_1, the database D queries the information to see if there was matching information $\{PID_R, K_R\}$. If the matching was successful, the database recognizes that the reader R was successful, that was $D \mid\equiv \phi(PID_R, K_R)$, the target D_1 was implemented. could be obtained by the message M_2, the reader R to obtain the group tag message from the database, available $R \in K_T$ (1), and in accordance with the agreement assumption $X_1(T \in K_T)$, we could know $R| \equiv R \overset{K}{\longleftrightarrow} T$ (2);

From the message M_4, we get $R \triangleleft *\mathrm{g}_{K_T}(PID_T \oplus K_T \oplus N_1)$, that was $R \triangleleft *(PID_T \oplus K_T \oplus N_1)_{K_T}$ (3), and the reader believed the freshness of the $\mathrm{g}_{K_T}(PID_T \oplus K_T \oplus N_1)$ message, that was $R| \equiv \#(PID_T \oplus K_T \oplus N_1)$. and then according to the freshness rule F_1, we could get $R| \equiv \#((PID_T \oplus K_T \oplus N_1), K_T)$ (4). According to the initialization assumptions X_4 $(R \in N_1)$ and $X_6(R| \equiv \phi(PID_T \oplus K_T \oplus N_2))$, we could see $R| \equiv \phi(PID_T \oplus K_T \oplus N_1)$ (5). So by the equations (1)-(5) and the message interpretation rule I_1, you could get equation $R| \equiv T \sim (PID_T \oplus K_T \oplus N_1)_{K_T}$, and the transformation could get $R| \equiv T \sim g_{K_T}(PID_T \oplus K_T \oplus N_1)$ (6).

Finally, based on Eqn. (6) and the communication message M_4, $R| \equiv T \sim \#g_{K_T}(PID_T \oplus K_T \oplus N_1)$ could be obtained and the target D_2 was verified.

The target D_3 proved that the process was the same as the target D_2, and the proof was not repeated.

According to the initial hypothesis $X3$, $D \in K_g$ (7) was available, and because the background database stored all the reader and tag information, $D| \equiv D \overset{K}{\longleftrightarrow} R$ (8) was obtained.

The message M_7 could be obtained $D \triangleleft *(P)_{K_g}$ (9), and according to the belief of the database to the freshness of the message M_7, $D| \equiv \#(P)$ could be obtained, and then according to the freshness rule F_1, $D| \equiv \#(P, K_g)$ (10) could be obtained.

According to the initial hypothesis X_3, $D \in (P)$ was owned by the rule P_1 and the message M_7, and $D| \equiv \phi(P)$ (11) was obtained according to the recognizable rule R_6.By (7)-(11) formula and message interpretation rules I_1, finally get $D \mid\equiv R \mid\sim (P)_{K_g}$, target D_4 get evidence.

In summary, the four goals of the improvement agreement had been verified.

5 Improved Protocol Security Analysis

5.1 Mutual Authentication Mechanism

The mutual authentication mechanism between the reader and the tag was the basis. The reader authenticated the message D to verify the legitimacy of the tag, and the tag authenticated the message F to verify the legitimacy of the reader. In addition, the database first authenticated the message A sent from the reader. After the authentication passed, the database generated an authorization identifier $Mark$, which was then transmitted to the reader and the reader decrypted and retained. Once the reader was impersonated, the grouping proof generated would be wrong when the $Mark$ identifier was unacquirable, so the database would fail authentication, and the attacker would fail.

5.2 Replay Attack

In most cases, the attackers would unexceptionally disguised as a reader to replay attack on tags, the process was improved as follows: The attacker eavesdroped on the communication channel, and acquired the communication data B, C, and then masqueraded as a normal reader to replay the intercepted messages B' and C' to the tag.

The tag used its own stored information to calculate whether the comparisons B and B' were equal. If equal, the tags were activated. However, when the tag obtained the random number N_1' sent from the attacker, the generated verification data D and the random number encrypted data E were sent to the attacker.

Since the attacker could not know the secret information PID_T, K_i, K_g and could not calculate and generate the correct response message F, G, the tag authentication failed and the protocol terminated.

Therefore, this protocol could resist replay attacks.

5.3 Brute-force Attack

Reference [3] relied solely on the selected encryption algorithm to ensure the security of the protocol, which was not rigorous. Therefore, compared with the Reference [3], this protocol encrypted all publicly transmitted random numbers to prevent attackers from eavesdropping on the obtained communication data and performing brute force attacks. Because in general cases, the label group key was already written, and the third party could not know it, and the random numbers in messages C, E, and G ($C = K_g \oplus N_1$, $E = K_g \oplus N_2$ and $G = K_g \oplus N_3$) were encrypted and transmitted by using the group key. The internal algorithms and data in messages C, E, and G were not available to the attacker. Then the attacker could not obtain the exact value of the random number to decrypt the authenticated messages D and E($D = \mathrm{g}_{K_{Ti}}(PID_{Ti} \oplus K_{Ti} \oplus N_1)$, $E = K_g \oplus N_2$). Therefore, this agreement could resist brute force attacks.

5.4 Impersonation Attack

An attacker could impersonate a reader or tag to attack. In the case of attacker pretending to be a reader, the reader and the database were wired securely during the authorization phase, so the attacker could not obtain the reader's identity and key information. The database pre-authentication failed, the protocol terminated, and the attack failed. Even if the attacker finally stole message A under secure wired communication and finally passed the authorization, due to the failure of the attacker to obtain K_g, it still could not decrypt the authorization ID mark under the encryption condition, so the correct group label authentication message P could not be generated. The authentication failed and the agreement terminated. If the attacker impersonates a legitimate tag, although the communication information B, C, F, and G could be obtained, since all the information was transmitted encrypted, the attacker could not obtain any of the PID_T, K_i, and K_g data. This meant that the correct random number N_1 could not be obtained by decrypting B, C, F, and G. In the end, the reader could not be provided with the correct data to be verified D, and the legal reader authentication failed and the protocol terminated. Therefore, this protocol could resist replay attacks.

5.5 Tracking Attack

In this agreement, although the tag ID was not changed, the tag pseudonym ID was used instead of the real ID for calculation during the entire protocol communication process. And the pseudonym and key information PID_T, K_i, K_g were updated and stored by the random number N_3 after the communication in each round. Each round of authentication of tag and reader also generated random numbers N_1 and N_2, which made this protocol somewhat fresh and unpredictable; therefore, the attacker could not trace the identity information of the tag only through eavesdropping or interception, and so on.

5.6 Proof of Forgery Attack

If an attacker had to forge a valid tag grouping proof to pass the final verification of the database, then it was necessary to:

Forge all valid grouping proof factors, that was, to obtain the pseudonym and private key information (PID_T and K_i) for all tags. However, it had been proven in the above attack statement that the probability of obtaining pseudonyms and private key information for all tags was impracticable; at the same time, it was also necessary to obtain the value of the random number N_3. However, in this protocol, N_3 encrypts the entire transmission, and the attacker could not crack the key without knowing the K_g key.

Get the database authorization logo, but the database and the reader were wired and securely transmitted. Even if it was insecure, the transmission of $Mark$ messages was also performed using multiple bit operations and random number encryption. Therefore, even when the attacker faked the reader, without knowing $Mark$, the attack was not likely to succeed. Therefore, this agreement could resist the proof of forgery attack.

Table 2: Security comparison of related protocols

Type of attacks	[2]	[3]	[10]	[11]	This article
No trusted third party	×	×	×	×	✓
Mutual authentication	✓	✓	×	✓	✓
Replay attack	×	×	✓	✓	✓
Brute-force attack	✓	✓	✓	×	✓
Impersonation attack	✓	✓	✓	✓	✓
Tracking attack	×	✓	✓	×	✓
Proof of forgery attack	✓	✓	×	×	✓

In summary, Table 2 gave a comparison of the security of this protocol and other RFID tag grouping proof protocols. Among them, ✓ meant that this type of attack could be resisted, and × meant that it could not resist this type of attack.

6 Performance Analysis

This section mainly analyzed the performance of this protocol in terms of the amount of tag calculations, storage capacity, and traffic volume. In accordance with the Gen-2 standard, the complexity and overhead of heavyweight encryption algorithms such as hash functions and elliptic ECC functions were significantly higher than those of pseudo-random functions, and the computational overhead required for lightweight MAC operations and pseudo-random encryption algorithms was comparable. [14]. See Table 3 for the performance of related protocols.

Tag computation overhead: The protocol tag side contained only lightweight pseudo-random function operations that satisfied the low-cost requirements of the Gen-2 standard and a simple bit operation (XOR operation), and only XOR operation was performed during the authentication process.

Tag communication overhead: In each round of communication, the single tag in the group conducted mainly two communications of D, E, and m_i.

Tag storage overhead: In the protocol initialization process, the single tag in the group stored tag pseudonyms, keys, and group key information $\{PID_T, K_i, K_g\}$.

Table 3: Performance comparison of related protocol tag

Related agreements	Computation overhead	Communication overhead	Storage overhead
[2]	$3M()$	$2I$	L
[3]	$H() + E() + 2X$	$3I$	$2L$
[10]	$3H() + 3M() + 3X$	$2I$	L
[11]	$2H() + 4g()$	$2I$	$2L$
This article	$4X + 5g()$	$3I$	$2L$

In Table 3, $H()$ denoted a hash operation, X denoted an exclusive-OR operation, $g()$ denoted a pseudo-random operation, $M()$ denoted a MAC operation, $E()$ denoted an elliptic curve operation, and the unit length of an ID and a key were both I, the unit of single communication overhead was L.

7 Conclusions

This paper proposed a lightweight, fully-certified RFID tag grouping proof protocol that met the Gen-2 standard. The protocol did not require the verification support of a trusted third party. It only required tag to support one-way pseudo-random function operations and simple XOR operations. Based on the reader and tag authentication, the database authorization identifier Mark was introduced to further encrypt and verify the generated tag grouping proof. GNY formal logic proved that the agreement was feasible and complete; after attack description analysis, the protocol met tag untraceability and could resist tag and reader impersonation attack and message replay attack. The protocol used encrypted random numbers and Mark authorization token to resist the tag proof of forgery attack and rapid brute force attack. Finally, according to the comparison of tag calculation, communication and storage overhead between the related protocols, it was proved that this protocol was better than the other tag grouping proof protocols presented. The next step was to add tag collision to the protocol for further study.

Acknowledgments

This work was supported in part by the Natural Science Foundation of China under Grant 51404216.

References

[1] H. Y. Chien, C. C. Yang, T. C. Wu, and C. F. Lee, "Two RFID-based solutions to enhance inpatient medication safety," *Journal of Medical Systems*, vol. 35, no. 3, pp. 369–375, 2011.

[2] Z. S. Da and G. Z. Ze, "Improved RFID yoking proof protocol," *Computer Engineering and Design*, vol. 38, no. 8, pp. 2076–2080, 2017.

[3] Y. M. Guo, S. D. Li, Z. H. Chen, and X. Liu, "A lightweight privacy-preserving grouping proof protocol for RFID systems," *Acta Electronica Sinica*, vol. 43, no. 2, pp. 289–292, 2015.

[4] H. F. Hong and H. L. Tian, *Privacy Protection Security Protocol Study (In Chinese)*, Beijing: Science Press, 2015.

[5] A. Juels, ""Yoking-proofs" for RFID tags," in *Proceedings of the Second IEEE Annual Conference on Pervasive Computing and Communications Workshops*, pp. 138–143, Mar. 2004.

[6] Z. Z. Kai, P. Tao, A. Liang, M G, and *et al.*, "High-reliable RFID grouping tag proof protocol," *Computer Engineering and Design*, vol. 39, no. 2, pp. 150–154, 2013.

[7] T. Korak, T. Plos, and A. Zankl, "Minimizing the costs of side-channel analysis resistance evaluations in early design steps," in *Eighth International Conference on Availability, Reliability and Security (ARES'13)*, pp. 169–177, 2013.

[8] Q. Lin and F. Zhang, "Ecc-based grouping-proof RFID for inpatient medication safety," *Journal of Medical Systems*, vol. 36, no. 6, pp. 3527–3531, 2012.

[9] J. Liu, M. Chen, B. Xiao, F. Zhu, S. Chen, and L. Chen, "Efficient RFID grouping protocols," *IEEE/ACM transactions on networking*, no. 5, pp. 3177–3190, 2016.

[10] M. Safkhani, N. Bagheri, M. Hosseinzadeh, M. Eslamnezhad Namin, and S. Rostampour, "On the security of an RFID-based parking lot management system," *International Journal of Communication Systems*, vol. 30, no. 15, p. e3313, 2017.

[11] 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.

[12] 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.

[13] 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.

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

[16] Z. Z. Wen, B. W. Li, F. L. Yi, and *et al.*, *Security Protocol Design and Analysis (In Chinese)*, Beijing: National Defense Industry Press, 2015.

[17] Z. Yang, G. P. Chang, and F. D. Hong, "A disordered and anonymous RFID grouping proof scheme," *Computer Engineering*, vol. 38, no. 20, pp. 85–88, 2012.

[18] B. Yuan and J. Liu, "A universally composable secure grouping-proof protocol for RFID tags," *Concurrency and Computation: Practice and Experience*, vol. 28, no. 6, pp. 1872–1883, 2016.

[19] Q. Zhang, X. Hu, J. Wei, and W. Liu, "Universally composable three-party password authenticated key exchange," in *International Conference on Cloud Computing and Security*, pp. 123–137, 2017.

Biography

Gao-feng Shen received his master degree in computer specialty from Huazhong University of Science and Technology (China) in June 2005. He is a lecturer in computer science in School Of Computer and Communication Engineering, Zhengzhou University of Light Industry. His current research interest fields include algorithm design, database and its application, data mining.

Shu-min Gu received her master degree in computation mathematics from Henan Normal University (China) in June 2007. She is an adjunct professor in mathematics in school of information and business, Zhongyuan University of Technology. Her current research interest fields include numerical solution of differential equation.

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

Cryptanalysis and Improvement of a User Authentication Scheme for Internet of Things Using Elliptic Curve Cryptography

Majid Bayat[1], Mohammad Beheshti Atashgah[2], Morteza Barari[2], and Mohammad Reza Aref[3]
(Corresponding author: Majid Bayat)
Department of Computer Engineering, Shahed University, Tehran, Iran[1]
ICT Complex, Malek-Ashtar University of Technology Tehran, Iran[2]
Department of Electrical Engineering, Sharif University of Technology[3]
(Email: mbayat@shahed.ac.ir)
(Received July 10, 2018; Revised and Accepted Dec. 22, 2018; First Online June 16, 2019)

Abstract

The concept of Internet of Things (IoT) is that objects and things via the Internet infrastructure can interconnect into a global dynamic extended network. In order to catch the final goal, IoT takes advantages of other useful technologies like RFIDs, WSNs, M2M communications, big data and cloud computing. Wireless Sensor Networks (WSNs) is one of the main parts of IoT's building blocks which can be used in almost all scopes of the IoT's applications. Because of the importance of the WSN's security, researchers are already working on new and efficient techniques on its different security schemes and protocols such as user authentication schemes. Recently, Wu *et al.* proposed a new user authentication scheme for Internet of Things-based wireless sensor networks. The scheme suggests a new method in which a user of IoT can be authenticated with a sensor node of the WSN through a communication with a gateway. Unfortunately, we have found that Wu *et al.*'s scheme has some security vulnerabilities and is not immune to some security attacks. This paper focuses on eliminating the security vulnerabilities of Wu *et al.*'s scheme by suggesting an enhanced scheme. We introduce a provable security for our scheme and present its formal security analysis by ProVerif. Moreover, we compare the proposed scheme with some other related schemes for WSNs in aspects of efficiency and security.

Keywords: Authentication; Internet of Things; ProVerif; Security; WSN

1 Introduction

The Internet of Things (IoT) is defined as a network of highly connected things and devices. In current perspective, the IoT includes various kinds of things, *e.g.*, sensors, actuators, RFID tags, smart phones or backend servers, which are very different in terms of size, capability and functionality. In other words, Internet of Things uses some technologies such as: Wireless Sensor Networks (WSN), Radio Frequency Identification (RFID), Machine-to-Machine communication, cloud computing and *etc.* According to Gartner's forecast [27], the IoT, which excludes PCs, smart phones and tablets, will grow to more than 26 billion units installed in 2020.

WSNs are crucial for the future of Internet of Things because it covers necessary IoT applications. The WSN contains small, wireless, ad-hoc sensor nodes which are used in a wide range of application scenarios such as health care, smart homes, military, environment and *etc.* [1, 8, 11, 15, 16, 19–21, 25].

Wireless sensor networks include three main parts: the users, the sensors and the gateway. The most important part is the gateway which can communicate with all the sensors. The gateway is accountable for the wireless sensor network security. The sensors and users register on relative gateway. Users who want to use from the data collected by the sensors should contact the gateway. Here, a common method is use of an session encryption key. Constructing a secure session key between the sensor and the user is a basic issue. If a user requests a data from a sensor of WSN, first of all, he/she should be identified for the legitimate access. The usual method is utilizing an authentication scheme among the sensors and users. So, authentication protocols are essential for WSN.

2 Related Works

In recent years, WSNs and their different security mechanisms have attracted many researcher's attention. Due to the limited resource of the WSNs, classic security mechanisms are not applicable because of much energy consumption. Therefore, many lightweight security methods are proposed for WSN (*e.g.* intrusion detection, secure data aggregation, secure and efficient routing proto-

cols, *etc.*) [23, 24, 30, 40].

Watero *et al.* proposed a user authentication protocol for WSN based on RSA in 2004 [34]. But, in 2009, Das showed that Watero *et al.*'s scheme is vulnerable against sensor forgery attack [7]. Moreover, he presented an other efficient authentication protocol that using smart card. But in 2010, his proposed scheme was evaluated by Chen *et al.* [5], He *et al.* [13], Khan *et al.* [17] and Vaidya *et al.* [33], respectively and it became clear that his scheme suffers from several security weaknesses like destitute of mutual authentication, the impersonation attack and the insider attack. Furthermore, Vaidya *et al.* showed that the Khan *et al.*'s scheme was also vulnerable against stolen smart card and the sensor nodes capture attacks and finally, they proposed an improved scheme. In 2011, Kumar *et al.* pointed out that He *et al.* [13] was vulnerable against information leakage attack and their scheme could not satisfy the following security properties: user anonymity, mutual authentication and constructing a shared session key by the sensor and the user [18].

Because of acceptable computational complexity, Elliptic Curve Cryptography (ECC) has been recently used for WSNs [2, 12, 22, 26, 29]. In 2011, Yeh *et al.* [38] showed that the Chen *et al.*'s scheme [5] suffers from the insider attack and lack of a password change phase. They also proposed the first ECC-based authentication scheme for WSNs. But, in 2011, Han [39] pointed out that Yeh *et al.*'s scheme does not satisfy forward security and mutual authentication. In 2013, Shai *et al.* [31] showed a two factor ECC-based authentication scheme. But, in 2014, Choi *et al.* presented that the Shai *et al.*'s scheme is not immune against the known session key attack and the off-line password guessing attack [6]. In addition, they presented a novel scheme. In 2015, Wu *et al.* [35] stated that the Choi *et al.*'s scheme still has some vulnerabilities such as user forgery attacks and off-line password guessing. Additionally, the user identity is revealed in the message and therefore, the privacy of user's identity is not met.

In order to pass the popular attacks, Turkanovic suggested a new scheme for heterogeneous wireless sensor networks in 2014 [32]. But, in 2015, Farash *et al.* [10] and Chang *et al.* [4] independently showed that Turkanović is vulnerable against the off-line password guessing and stolen verifier attacks. Moreover, in their scheme, the identity of the user can be traced.

In 2014, Hsieh *et al.* [14] showed that Vaidya *et al.*'s scheme [33] is vulnerable to off-line password guessing attack and the insider attack. Additionally, they presented a new scheme in their paper.

Wu *et al.* [36] presented a new scheme for WSNs which is based on the Fantacci *et al.* [9] and Nguyen [28] recommendations for IoT security. In this scheme, a user sends messages to a gateway at first and after that the gateway communicates with a sensor. Finally, by the Wu *et al.*'s scheme a user, a gateway and a sensor can authenticate each other.

In this paper, we show that the Wu *et al.*'s scheme is vulnerable to some security weaknesses and to overcome those flaws, we suggest an enhanced authentication scheme.

2.1 Our Contribution

In this paper, we show that the Wu *et al.*'s user authentication scheme [36] is not a secure scheme because it is vulnerable against forgery and Denial of Service (DoS) attacks. After that, in order to eliminate the weaknesses we suggest an enhanced user authentication scheme for IoT. In addition, we present a formal security analysis by ProVerif and a provable security in the random oracle model for our scheme. Finally, we compare the proposed scheme with related schemes in case of security and efficiency. The results indicate that our scheme is a suitable and practical design for utilizing in IoT.

2.2 Paper Organization

The rest of this paper is organized as follows: We review Wu *et al.*'s scheme and its security analysis in Section 2. In Section 3, we introduce our improved scheme. The security analysis of the proposed scheme and some comparisons are posed in Section 4. Finally, we conclude the paper in Section 5.

3 Review of the Wu *et al.*'s Scheme

In this section, we review the Wu *et al.*'s scheme [36]. Their scheme includes four phases: Initialization, Registration, Login and Authentication. Table 1 presents utilized notations of the Wu *et al.*'s scheme.

3.1 Initialization

GW obtains an addition group G with a large prime order q on $E(F_q)$. P is a generator of group G. ID_{GW} is the identity of GW. GW also picks a secret key x and two hash functions $h(\cdot)$ and $h_1(\cdot)$.

3.2 Registration

This phase includes registration procedures for user U_i and sensor S_j.
For U_i:

1) U_i picks a random number r_0, his/her own identity ID_i and a password PW_i. After that, he/she computes $MP_i = h(r_0 \parallel PW_i)$ and $MI_i = h(r_0 \parallel ID_i)$, and sends $\{MP_i, MI_i, ID_i\}$ to GW through a secure channel.

2) GW computes $e_i = h(ID_{GW} \parallel x \parallel MI_i) \oplus MP_i$ and $f_i = h(MI_i \parallel x) \oplus MI_i$. GW injects (e_i, f_i, P, p, q) into the smart card, saves ID_i in the database for auditing, and gives the smart card to U_i by a secure channel.

Table 1: Symbols were used in the Wu *et al.*'s and proposed schemes

Symbols	Description
p, q	Large prime numbers
$E(F_q)$	An elliptic curve E over the finite field F_q
G	An additive subgroup of points of E with order q
P	A generator of G
GW, x	The gateway and its corresponding secret key
U_i, ID_i, PW_i	The i-th user, his/her identity and password
S_j, SID_j	The j-th sensor and its identity
sk_u, sk_s	The session keys computed by the user and the sensor
A	The adversary (malicious)
$h(.), h_1(.)$	One-way hash functions
T_i	Timestamp of user U_i
l	Security parameter of system
$E_k(.)/D_k(.)$	The symmetric encryption/decryption function with key k
$a \oplus b, a\|b$	The XOR operation and the conjuction with string a and b
$a =?b$	Check whether a equal b

3) U_i saves $d_i = h(ID_i \parallel PW_i) \oplus r_0$ into the smart card.

For S_j:

1) S_j submits SID_j to GW through a secure channel.

2) GW calculates $c_j = h(SID_j \parallel x)$ and sends it to S_j through a secure channel. S_j stores SID_j and c_j.

In addition, if a sensor be substituted by the other sensor one or a new sensor connects the WSN, the new sensor should register to GW similar to the upper steps.

3.3 Login and Authentication

1) U_i inserts his/her card and enters ID_i and PW_i. $r_1 = d_i \oplus h(ID_i \parallel PW_i)$, $MI_i = h(r_1 \parallel ID_i)$ and $\mathrm{MP}_i = h(r_1 \parallel PW_i)$ are computed by the smart card.

2) U_i picks a random number $\alpha \in [1, q-1]$, r_2 and r_3. U_i obtains the sensor S_j as the partner and calculates $MI_i^{new} = h(r_2 \parallel ID_i)$, $B_1 = e_i \oplus MP_i \oplus r_3$, $B_2 = \alpha P$, $B_3 = f_i \oplus MI_i \oplus MI_i^{new} \oplus h(r_3 \parallel MI_i)$, $B_4 = h(r_3 \parallel MI_i^{new} \parallel B_2) \oplus ID_i$ and $B_5 = h(ID_i \parallel MI_i \parallel MI_i^{new} \parallel SID_j)$. Then, he/she sends $M_1 = \{MI_i, SID_j, B_1, B_2, B_3, B_4, B_5\}$ to S_j.

3) GW computes $r_3 = B_1 \oplus h(ID_{GW} \parallel x \parallel MI_i)$, $MI_i^{new} = B_3 \oplus h(MI_i \parallel x) \oplus h(r_3 \parallel MI_i)$ and $ID_i = B_4 \oplus h(r_3 \parallel MI_i^{new} \parallel B_2)$. Then, GW checks if ID_i is in database and $B_5 =?h(ID_i \parallel MI_i \parallel MI_i^{new} \parallel SID_j)$. If they hold, GW calculates $c_j = h(SID_j \parallel x)$ and $D_1 = h(MI_i \parallel SID_j \parallel c_j \parallel B_2)$. Next, the message $M_2 = \{MI_i, SID_j, B_2, D_1\}$ is sent to sensor S_j.

4) S_j checks SID_j and $D_1 \stackrel{?}{=} h(MI_i \parallel SID_j \parallel c_j \parallel B_2)$. If they are incorrect, S_j fails the session. Otherwise, S_j picks a random $\beta \in [1, q-1]$ and then computes $C_1 = \beta P$, $C_2 = \beta B_2$, $sk_s = h_1(B_2 \parallel C_1 \parallel C_2)$, $C_3 = h(MI_i \parallel SID_j \parallel sk_s)$ and $C_4 = h(c_j \parallel MI_i \parallel SID_j)$. Next, S_j sends $M_3 = \{C_1, C_3, C_4\}$ to GW.

5) GW checks $C_4 \stackrel{?}{=} h(c_j \parallel MI_i \parallel SID_j)$. If it holds, then GW calculates $D_2 = h(ID_{GW} \parallel x \parallel MI_i^{new}) \oplus h(MI_i^{new} \parallel r_3)$, $D_3 = h(MI_i^{new} \parallel x) \oplus h(MI_i \parallel r_3)$ and $D_4 = h(ID_i \parallel MI_i \parallel MI_i^{new} \parallel SID_j \parallel D_2 \parallel D_3 \parallel r_3)$. Finally, GW sends $M_4 = \{C_1, C_3, D_2, D_3, D_4\}$ to U_i.

6) U_i checks $D_4 \stackrel{?}{=} h(ID_i \parallel MI_i \parallel MI_i^{new} \parallel SID_j \parallel D_2 \parallel D_3 \parallel r_3)$. If it holds, U_i computes $B_6 = \alpha C_1$ and $sk_u = h_1(B_2 \parallel C_1 \parallel B_6)$. After that, U_i checks whether $C_4 \stackrel{?}{=} h(MI_i \parallel SID_j \parallel sk_u)$. If it holds, the smart card calculates a new data $d_i^{new} = r_2 \oplus h(ID_i \parallel PW_i)$, $e_i^{new} = D_2 \oplus h(MI_i^{new} \parallel r_3) \oplus h(r_2 \parallel PW_i)$, and $f_i^{new} = D_3 \oplus MI_i^{new} \oplus h(MI_i \parallel r_3)$. Finally, it replaces (d_i, e_i, f_i) with $(d_i^{new}, e_i^{new}, f_i^{new})$, respectively.

3.4 Password Change

1) This step is identical with the step 1 of login and authentication phase.

2) U_i randomly picks values r_4 and r_5 and then computes $MI_i^{new} = h(r_4 \parallel ID_i)$, $B_7 = e_i \oplus MP_i \oplus r_5$, $B_8 = f_i \oplus MI_i \oplus MI_i^{new} \oplus h(r_5 \parallel MI_i)$, $B_9 = ID_i \oplus h(r_5 \parallel MI_i^{new} \parallel B_2)$ and $B_{10} = h(ID_i \parallel MI_i \parallel MI_i^{new} \parallel r_5)$

3) GW calculates $r_5 = B_7 \oplus h(ID_{GW} \parallel x \parallel MI_i)$, $MI_i^{new} = B_8 \oplus h(MI_i \parallel x) \oplus h(r_3 \parallel MI_i)$ and $ID_i = B_9 \oplus h(r_5 \parallel MI_i^{new} \parallel B_2)$, and checks the validity of ID_i and $B_{10} =?h(ID_i \parallel MI_i \parallel MI_i^{new} \parallel r_5)$. If either of them is failed, the request is rejected. Otherwise, GW computes $D_5 = h(ID_{GW} \parallel x \parallel MI_i^{new}) \oplus h(MI_i^{new} \parallel r_5)$, $D_6 = h(MI_i^{new} \parallel x) \oplus h(MI_i \parallel r_5)$ and $D_7 = h(ID_i \parallel r_5 \parallel MI_i \parallel MI_i^{new} \parallel D_5 \parallel D_6)$. GW sends $M_6 = \{D_5, D_6, D_7\}$ to the user U_i.

4) U_i checks $D_7 \stackrel{?}{=} h(ID_i \parallel r_5 \parallel MI_i \parallel MI_i^{new} \parallel D_5 \parallel D_6)$. If this equation does not hold, U_i fails the session. Otherwise, U_i is asked to input a new password PW_i^{new}. Then, the smart card calculates $MP_i^{new} = h(r_4 \parallel PW_i^{new})$, $e_i^{new2} = D_5 \oplus h(MI_i^{new} \parallel r_5) \oplus MP_i^{new}$, $f_i^{new2} = D_6 \oplus h(MI_i \parallel r_5) \oplus MI_i^{new}$ and $d_i^{new2} = r_4 \oplus h(ID_i \parallel PW_i^{new})$, and finally updates (d_i, e_i, f_i) with $\left(d_i^{new2}, e_i^{new2}, f_i^{new2}\right)$.

3.5 Security Analysis of Wu *et al.*'s Scheme

In this section, we show that Wu *et al.*'s scheme is vulnerable against two types of attacks: Denial of Service (DoS) attack and forgery attack.

Table 2: Login and Authentication phases of the Wu *et al.*'s scheme

U_i	GW	S_j
Step One: input ID_i, PW_i compute $r_1 = d_i \oplus h(ID_i \parallel PW_i)$ $\mathrm{MI}_i = h(r_1 \parallel ID_i)$ and $\mathrm{MP}_i = h(r_1 \parallel PW_i)$ choose random numbers $\alpha \in [1, q-1]$, r_2 and r_3 compute the followings: $MI_i^{new} = h(r_2 \parallel ID_i)$ $B_1 = e_i \oplus MP_i \oplus r_3$ $B_2 = \alpha P$ $B_3 = f_i \oplus MI_i \oplus MI_i^{new} \oplus h(r_3 \parallel MI_i)$ $B_4 = h(r_3 \parallel MI_i^{new} \parallel B_2) \oplus ID_i$ $B_5 = h(ID_i \parallel MI_i \parallel MI_i^{new} \parallel SID_j)$ $\xrightarrow{M_1=\{MI_i, SID_j, B_1, B_2, B_3, B_4, B_5\}}$		
	Step Two: compute the followings: $r_3 = B_1 \oplus h(ID_{GW} \parallel x \parallel MI_i)$ $MI_i^{new} = B_3 \oplus h(MI_i \parallel x) \oplus h(r_3 \parallel MI_i)$ $ID_i = B_4 \oplus h(r_3 \parallel MI_i^{new} \parallel B_2)$ check: ID_i, $B_5 \stackrel{?}{=} h(ID_i \parallel MI_i \parallel MI_i^{new} \parallel SID_j)$ compute: $c_j = h(SID_j \parallel x)$ $D_1 = h(MI_i \parallel SID_j \parallel c_j \parallel B_2)$ $\xrightarrow{M_2=\{MI_i, SID_j, B_2, D_1\}}$	
		Step Three: check: SID_j check: ID_i, $D_1 \stackrel{?}{=} h(MI_i \parallel SID_j \parallel c_j \parallel B_2)$ choose random $\beta \in [1, q-1]$ compute the followings: $C_1 = \beta P$ $C_2 = \beta B_2$ $sk_s = h_1(B_2 \parallel C_1 \parallel C_2)$ $C_3 = h(MI_i \parallel SID_j \parallel sk_s)$ $C_4 = h(c_j \parallel MI_i \parallel SID_j)$ $\xleftarrow{M_3=\{C_1, C_3, C_4\}}$
	Step Four: check: $C_4 \stackrel{?}{=} h(c_j \parallel MI_i \parallel SID_j)$ compute the followings: $D_2 = h(ID_{GW} \parallel x \parallel MI_i^{new}) \oplus h(MI_i^{new} \parallel r_3)$ $D_3 = h(MI_i^{new} \parallel x) \oplus h(MI_i \parallel r_3)$ $D_4 = h(ID_i \parallel MI_i \parallel MI_i^{new} \parallel SID_j \parallel D_2 \parallel D_3 \parallel r_3)$ $\xleftarrow{M_4=\{C_1, C_3, D_2, D_3, D_4\}}$	
Step Five: check: $D_4 \stackrel{?}{=} h(ID_i \parallel MI_i \parallel MI_i^{new} \parallel SID_j \parallel D_2 \parallel D_3 \parallel r_3)$ compute the followings: $B_6 = \alpha C_1$ $sk_u = h_1(B_2 \parallel C_1 \parallel B_6)$ check: $C_4 \stackrel{?}{=} h(MI_i \parallel SID_j \parallel sk_u)$ compute: $d_i^{new} = r_2 \oplus h(ID_i \parallel PW_i)$ $e_i^{new} = D_2 \oplus h(MI_i^{new} \parallel r_3) \oplus h(r_2 \parallel PW_i)$ $f_i^{new} = D_3 \oplus MI_i^{new} \oplus h(MI_i \parallel r_3)$ replace (d_i, e_i, f_i) with $(d_i^{new}, e_i^{new}, f_i^{new})$		

- **Denial of service attack:** An attacker can masquerade himself/herself as a real user U_i and apply DoS attack against server GW. Since the term $M_1 = \{MI_i, SID_j, B_1, B_2, B_3, B_4, B_5\}$ is always valid, an attacker can apply DoS attack by sending this message to the GW. Note that $M_1 = \{MI_i, SID_j, B_1, B_2, B_3, B_4, B_5\}$ does not contain any fresh term like a time stamp, the attacker can frequently send M_i to the GW and finally, this action allows the server GW to be unavailable. Moreover, the attacker can provide DoS attack more effectively by using Distributed Denial of Service (DDoS) attack.

- **Forgery attack:** Although Wu *et al.*'s stated that their proposed scheme is immune to user forgery attack, but we show that an adversary can play the role of a user U_i and a sensor S_j and consequently GW is convinced that U_i and S_j established a secure session key.

  The adversary records all messages M_1, M_2, M_3 and M_4 of a successful session between the U_i, S_j and GW. After that, the adversary starts a new session and sends the recorded $M_1 = \{MI_i, SID_j, B_1, B_2, B_3, B_4, B_5\}$ to server GW. Upon receiving M_1, GW executes its computations and verifications and sends generated M_2 to sensor S_j.

  The adversary intercepts M_2, chooses a random number β' and computes the following parameters:

$$\begin{aligned} C_1' &= \beta' P \\ C_2' &= \beta' B_2 \end{aligned}$$

  The attacker computes a new valid session key sk_s' and the value C_3' as follows:

$$\begin{aligned} sk_s' &= h_1(B_2 \parallel C_1' \parallel C_2') \\ C_3' &= h(MI_i \parallel SID_j \parallel sk_s') \end{aligned}$$

  The adversary uses the recorded value C_4 of the previous session and sends a new message M_3' to GW instead of sensor S_j.

$$M_3' = \{C_1', C_3', C_4\}$$

  Upon receiving M_3', GW verifies the value C_4 and accepts it as a valid value. GW generates the message M_4 and sends it to U_i. Therefore, the adversary can forge U_i and S_j and convince GW that S_j and U_i established a secure session key with each other.

  The proposed attack is arisen of two weaknesses. First, a valid submitted message M_1 in a session, is a valid message for GW at next sessions and second issue is that GW does not utilize a random number in its computations.

4 The Proposed Scheme

In this section, we propose a new scheme that solves the security problems of Wu *et al.*'s scheme. Like Wu *et al.*'s scheme, our new scheme includes four phases: Initialization, Registration, Login and Authentication, and Password change.

4.1 Initialization

GW firstly generates an addition group G with a large prime order q on $E(F_q)$. P is a generator of group G. ID_{GW} is the identity of GW. GW also picks a secret key x and two hash functions $h(\cdot)$ and $h_1(\cdot)$.

4.2 Registration

This phase includes registration procedures for user U_i and sensor S_j.
For U_i:

1) U_i chooses a number r_0 at random, his/her own identity ID_i and a password PW_i. After that, he/she computes the followings:

$$\begin{aligned} MP_i &= h(r_0 \parallel PW_i) \\ MI_i &= h(r_0 \parallel ID_i) \end{aligned} \quad (1)$$

   and then sends $\{MP_i, MI_i, ID_i\}$ to GW via a secure channel.

2) GW computes

$$e_i = h(ID_{GW} \parallel x \parallel MI_i) \oplus MP_i \quad (2)$$
$$f_i = h(MI_i \parallel x) \oplus MI_i \quad (3)$$

   Then, GW injects (e_i, f_i, P, p, q) into the smart card, saves ID_i in the database for auditing, and gives the card to U_i through a secure channel.

3) U_i saves the following d_i into the relative smart card.

$$d_i = h(ID_i \parallel PW_i) \oplus r_0$$

For S_j:

1) S_j submits SID_j to GW via a secure channel.

2) GW calculates $c_j = h(SID_j \parallel x)$ and sends it to S_j through a secure channel. Moreover, S_j stores the parameters SID_j and c_j.

4.3 Login and Authentication

1) U_i inserts his/her smart card and enters ID_i and PW_i. The card computes

$$\begin{aligned} r_1 &= d_i \oplus h(ID_i \parallel PW_i) \\ MI_i &= h(r_1 \parallel ID_i) \\ MP_i &= h(r_1 \parallel PW_i) \end{aligned}$$

2) U_i chooses random numbers $\alpha \in [1, q-1]$, r_2 and r_3, selects sensor S_j as the partner, obtains a time stamp T_i and calculates

$$\begin{aligned} MI_i^{new} &= h\left(r_2 \parallel ID_i\right) \\ B_1 &= e_i \oplus MP_i \oplus r_3 \\ B_2 &= \alpha P \\ B_3 &= f_i \oplus MI_i \oplus MI_i^{new} \oplus h\left(r_3 \parallel MI_i\right) \\ B_4 &= h\left(r_3 \parallel MI_i^{new} \parallel B_2\right) \oplus ID_i \\ B_5 &= h(ID_i \parallel MI_i \parallel MI_i^{new} \parallel SID_j \parallel T_i) \end{aligned}$$

Then, he/she sends M_1 to GW.

$$M_1 = \{MI_i, SID_j, B_1, B_2, B_3, B_4, B_5, T_i\}$$

3) GW checks whether $|T - T_i| < \Delta$, where T is current time and Δ is a predefined delay. If $|T - T_i| > \Delta$, GW rejects the session. If T_i is accepted, GW computes

$$\begin{aligned} r_3 &= B_1 \oplus h(ID_{GW} \parallel x \parallel MI_i) \\ MI_i^{new} &= B_3 \oplus h(MI_i \parallel x) \oplus h(r_3 \parallel MI_i) \\ ID_i &= B_4 \oplus h\left(r_3 \parallel MI_i^{new} \parallel B_2\right) \end{aligned}$$

Then, GW checks if ID_i is in database and $B_5 \stackrel{?}{=} h(ID_i \parallel MI_i \parallel MI_i^{new} \parallel SID_j \parallel T_i)$. If one of the verifications fails, the session is rejected. GW picks $\lambda \in [1, q-1]$ at random, obtains a time stamp T_G and calculates

$$\begin{aligned} C_0 &= \lambda P \\ c_j &= h\left(SID_j \parallel x\right) \\ D_1 &= h\left(MI_i \parallel SID_j \parallel c_j C_0 \parallel B_2 \parallel T_G\right) \end{aligned}$$

Next, the message M_2 is sent to sensor S_j.

$$M_2 = \{MI_i, SID_j, B_2, D_1, C_0, T_G\}$$

4) S_j checks SID_j, $|T - T_G| > \Delta$ and $D_1 \stackrel{?}{=} h(MI_i \parallel SID_j \parallel c_j C_0 \parallel B_2 \parallel T_G)$. If either checking fails, S_j rejects the session. Otherwise, S_j chooses a random $\beta \in [1, q-1]$ and computes

$$\begin{aligned} C_1 &= \beta P \\ C_2 &= \beta B_2 \\ sk_s &= h_1\left(B_2 \parallel C_1 \parallel C_2\right) \\ C_3 &= h\left(MI_i \parallel SID_j \parallel sk_s\right) \\ C_4 &= h\left(c_j C_0 \parallel MI_i \parallel SID_j\right) \end{aligned}$$

Next, S_j sends M_3 to GW.

$$M_3 = \{C_1, C_3, C_4\}$$

5) After receiving M_3, GW checks $C_4 \stackrel{?}{=} h(c_j C_0 \parallel MI_i \parallel SID_j)$. If it holds, GW computes

$$\begin{aligned} D_2 &= h(ID_{GW} \parallel x \parallel MI_i^{new}) \oplus h\left(MI_i^{new} \parallel r_3\right) \\ D_3 &= h\left(MI_i^{new} \parallel x\right) \oplus h\left(MI_i \parallel r_3\right) \\ D_4 &= h(ID_i \parallel MI_i \parallel MI_i^{new} \parallel SID_j \parallel D_2 \parallel D_3 \parallel r_3) \end{aligned}$$

Finally, GW sends M_4 to U_i.

$$M_4 = \{C_1, C_3, D_2, D_3, D_4\}$$

6) Upon receiving M_4, U_i checks $D_4 \stackrel{?}{=} h(ID_i \parallel MI_i \parallel MI_i^{new} \parallel SID_j \parallel D_2 \parallel D_3 \parallel r_3)$. If it is true, U_i computes

$$\begin{aligned} B_6 &= \alpha C_1 \\ sk_u &= h_1\left(B_2 \parallel C_1 \parallel B_6\right) \end{aligned}$$

After that, U_i checks $C_4 \stackrel{?}{=} h\left(MI_i \parallel SID_j \parallel sk_u\right)$. If it holds, the smart card calculates new data as follows

$$\begin{aligned} d_i^{new} &= r_2 \oplus h\left(ID_i \parallel PW_i\right) \\ e_i^{new} &= D_2 \oplus h\left(MI_i^{new} \parallel r_3\right) \oplus h(r_2 \parallel PW_i) \\ f_i^{new} &= D_3 \oplus MI_i^{new} \oplus h\left(MI_i \parallel r_3\right) \end{aligned}$$

Finally, it replaces (d_i, e_i, f_i) with $(d_i^{new}, e_i^{new}, f_i^{new})$, respectively. Table 3 presents the login and authentication phase.

4.4 Password Change

1) This step is identical with the Step 1 of login and authentication phase.

2) U_i randomly chooses values r_4 and r_5 and calculates the followings

$$\begin{aligned} MI_i^{new} &= h\left(r_4 \parallel ID_i\right) \\ B_7 &= e_i \oplus MP_i \oplus r_5 \\ B_8 &= f_i \oplus MI_i \oplus MI_i^{new} \oplus h\left(r_5 \parallel MI_i\right) \\ B_9 &= ID_i \oplus h\left(r_5 \parallel MI_i^{new} \parallel B_2\right) \\ B_{10} &= h(ID_i \parallel MI_i \parallel MI_i^{new} \parallel r_5) \end{aligned}$$

U_i sends $M_5 = \{M_i, B_7, B_8, B_9, B_{10}\}$ and a password change request to GW.

3) Upon receiving M_5 and the password change request, GW calculates

$$\begin{aligned} r_5 &= B_7 \oplus h(ID_{GW} \parallel x \parallel MI_i) \\ MI_i^{new} &= B_8 \oplus h\left(MI_i \parallel x\right) \oplus h\left(r_5 \parallel MI_i\right) \\ ID_i &= B_9 \oplus h\left(r_5 \parallel MI_i^{new} \parallel B_2\right) \end{aligned}$$

and then checks the validity of ID_i and also checks the following:

$$B_{10} = ?h\left(ID_i \parallel MI_i \parallel MI_i^{new} \parallel r_5\right)$$

If either of them fails, the request is rejected. Otherwise, GW computes

$$\begin{aligned} D_5 &= h(ID_{GW} \parallel x \parallel MI_i^{new}) \oplus h(MI_i^{new} \parallel r_5) \\ D_6 &= h\left(MI_i^{new} \parallel x\right) \oplus h\left(MI_i \parallel r_5\right) \\ D_7 &= h(ID_i \parallel r_5 \parallel MI_i \parallel MI_i^{new} \parallel D_5 \parallel D_6) \end{aligned}$$

GW sends $M_6 = \{D_5, D_6, D_7\}$ to the user U_i with grant.

4) After receiving M_6, U_i checks $D_7 =?h(ID_i \parallel r_5 \parallel MI_i \parallel MI_i^{new} \parallel D_5 \parallel D_6)$. If this equation is rejected, U_i fails the session. Otherwise, U_i is requested to input a new password PW_i^{new}. Then, the following values are computed by the smart card:

$$\begin{aligned} MP_i^{new} &= h\left(r_4 \parallel PW_i^{new}\right) \\ e_i^{new2} &= D_5 \oplus h\left(MI_i^{new} \parallel r_5\right) \oplus MP_i^{new} \\ f_i^{new2} &= D_6 \oplus h\left(MI_i \parallel r_5\right) \oplus MI_i^{new} \\ d_i^{new2} &= r_4 \oplus h\left(ID_i \parallel PW_i^{new}\right) \end{aligned}$$

Finally U_i, updates (d_i, e_i, f_i) with $(d_i^{new2}, e_i^{new2}, f_i^{new2})$, respectively.

5 Security Analysis

In this section, we evaluate the security of our scheme. We discuss the security properties of the proposed scheme and present a provable security of our scheme. In addition, a formal proof of the proposed scheme is introduced. Finally security and efficiency comparisons are posed.

5.1 Analysis of the Security Properties

- **Resistant to insider attack:** Within registration phase, U_i sends $MP_i = h\left(r_0 \parallel PW_i\right)$ to GW. The adversary is incapable to guess the correct password PW_i because the adversary has not the random r_0. Thus a malicious GW cannot obtain the password of users.

- **Resistant to off-line password guessing attack:** Assume an adversary A is eavesdropping the communications between U_i and GW to obtain the password PW_i. The adversary records message M_1 (**??**) and try to find the password. Since the password is not contained at the M_1, the adversary is unable to find PW_i. In addition, let the adversary steels the smart card and obtains e_i, f_i and d_i. Since the adversary has not r_0 and the secret value x, it cannot find the passwords via e_i and d_i. Thus the proposed protocol is immune to off-line password guessing attack.

- **Resistant to user forgery attack:** In order to forge U_i, the adversary A should generate a valid message M_1. Since A does not know x, it is unable to calculate valid values $B_1 = h\left(ID_{GW} \parallel x \parallel MI_i\right) \oplus r_3$ and $B_3 = h\left(MI_i \parallel x\right) \oplus MI_i^{new} \oplus h\left(r_3 \parallel MI_i\right)$. In addition, due to the used time stamp, the adversary cannot utilize an old message M_1 to forge U_i. Thus the proposed protocol is secure against user forgery attack.

- **Resistant to gateway forgery attack:** If the adversary A wants to forge GW, it should compute $D_1(20), D_2(28), D_3(29)$ and $r_3(15)$ correctly. Since A has not the secret value x, it is incapable to generates the needed values. Therefore, A is unable to forge GW in our scheme.

- **Resisitant to sensor capture attack:** Sensor capturing attack leads that using retrieved information from compromise sensor node to execute attacks in IoT environment. Adversary attempts to retrieve information about other sensor nodes, and the users in order to compromise any other secure communication between the users and the non-compromised sensor nodes in the IoT. In our scheme, each sensor has a unique identity SID_j and the corresponding secret value c_j. Thus, compromising a sensor does not affect on the other sensors.

- **Resistant to de-synchronization attack:** It implies that the legitimate user's login and authentication is rejected by the gateway. In the proposed scheme, the gateway checks the password in a session before password changing. This avoids inserting wrong passwords. Moreover, inappropriate data between the user and the gateway causes this attack. The gateway only saves the identity for audit and it does not store any data about the users. Data is changed on the user side. It is infeasible that inappropriate data become visible between the gateway and the user. Thus, the proposed scheme is immune to the de-synchronization attack.

- **Resistant to replay attack:** Due to the utilized random fresh numbers by user, gateway and sensor and usage of time stamp, our protocol is immune against reply attack.

- **Resistant to known-key attack:** In our scheme, the session key is $sk_s = h_1\left(B_2 \parallel C_1 \parallel C_2\right)$, where $C_2 = \beta B_2 = \alpha C_1$. Since β and α are randomly selected at each session, the session keys are completely independent. Thus, if A can obtain a session key, it cannot calculates the next session keys.

- **User anonymity:** The proposed protocol utilizes a pseudonym MI_i as the identity of U_i and it be updated in each authentication and password change phase. Therefore, the adversary cannot trace U_i via MI_i. In addition, MI_i does nor reveal ID_i because it is a hash result of ID_i and r_1. Thus, our scheme satisfies the anonymity property for user U_i.

- **Strong forward secrecy:** Assume the adversary who records the flows of previous sessions, obtains all secret information of U_i, S_j and GW. By assuming the intractability ECCDH problem, it cannot compute the the random values α (10) and β (22) and the session key of previous sessions. Thus, the proposed scheme satisfies strong forward secrecy.

Table 3: Login and Authentication phases of the proposed scheme

U_i	GW	S_j
Step 1: input ID_i, PW_i compute $r_1 = d_i \oplus h(ID_i \parallel PW_i)$ $\mathrm{MI}_i = h(r_1 \parallel ID_i)$ and $\mathrm{MP}_i = h(r_1 \parallel PW_i)$ choose random numbers $\alpha \in [1, q-1]$, r_2 and r_3 compute the followings: $MI_i^{new} = h(r_2 \parallel ID_i)$ $B_1 = e_i \oplus MP_i \oplus r_3$ $B_2 = \alpha P$ $B_3 = f_i \oplus MI_i \oplus MI_i^{new} \oplus h(r_3 \parallel MI_i)$ $B_4 = h(r_3 \parallel MI_i^{new} \parallel B_2) \oplus ID_i$ $B_5 = h(ID_i \parallel MI_i \parallel MI_i^{new} \parallel SID_j \parallel T_i)$ $\xrightarrow{M_1=\{MI_i, SID_j, B_1, B_2, B_3, B_4, B_5, T_i\}}$		
	Step 2: Verify T_i and compute the followings: $r_3 = B_1 \oplus h(ID_{GW} \parallel x \parallel MI_i)$ $MI_i^{new} = B_3 \oplus h(MI_i \parallel x) \oplus$ $h(r_3 \parallel MI_i)$ $ID_i = B_4 \oplus h(r_3 \parallel MI_i^{new} \parallel B_2)$ check: ID_i, $B_5 =? h(ID_i \parallel MI_i \parallel MI_i^{new} \parallel SID_j \parallel T_i)$ choose $\lambda \in [1, q-1]$ compute: $C_0 = \lambda P$ $c_j = h(SID_j \parallel x)$ $D_1 = h(MI_i \parallel SID_j \parallel c_j C_0 \parallel B_2 \parallel T_G)$ $\xrightarrow{M_2=\{MI_i, SID_j, B_2, D_1, C_0, T_G\}}$	
		Step 3: check T_G check SID_j check ID_i, $D_1 =? h(MI_i \parallel SID_j \parallel c_j C_0 \parallel B_2 \parallel T_G)$ choose random $\beta \in [1, q-1]$ compute the followings: $C_1 = \beta P$ $C_2 = \beta B_2$ $sk_s = h_1(B_2 \parallel C_1 \parallel C_2)$ $C_3 = h(MI_i \parallel SID_j \parallel sk_s)$ $C_4 = h(c_j C_0 \parallel MI_i \parallel SID_j)$ $\xleftarrow{M_3=\{C_1, C_3, C_4\}}$
	Step 4: check: $C_4 =? h(c_j C_0 \parallel MI_i \parallel SID_j)$ compute the followings: $D_2 = h(ID_{GW} \parallel x \parallel MI_i^{new}) \oplus h(MI_i^{new} \parallel r_3)$ $D_3 = h(MI_i^{new} \parallel x) \oplus h(MI_i \parallel r_3)$ $D_4 = h(ID_i \parallel MI_i \parallel MI_i^{new} \parallel SID_j \parallel D_2 \parallel D_3 \parallel r_3)$ $\xleftarrow{M_4=\{C_1, C_3, D_2, D_3, D_4\}}$	
Step 5: check: $D_4 =? h(ID_i \parallel MI_i \parallel MI_i^{new} \parallel SID_j \parallel D_2 \parallel D_3 \parallel r_3)$ compute the followings: $B_6 = \alpha C_1$ $sk_u = h_1(B_2 \parallel C_1 \parallel B_6)$ check: $C_4 =? h(MI_i \parallel SID_j \parallel sk_u)$ compute: $d_i^{new} = r_2 \oplus h(ID_i \parallel PW_i)$ $e_i^{new} = D_2 \oplus h(MI_i^{new} \parallel r_3) \oplus h(r_2 \parallel PW_i)$ $f_i^{new} = D_3 \oplus MI_i^{new} \oplus h(MI_i \parallel r_3)$ replace (d_i, e_i, f_i) with $(d_i^{new}, e_i^{new}, f_i^{new})$		

5.2 Provable Security

This section introduces the formal proof of our scheme based on the Bresson *et al.*'s model [3]. In the presented proof, The protocol P includes three entities; one user U, one sensor S and a gateway GW. The notation I is used for denoting different users.

We utilize U^i as the $i-th$ instance of U. GW^t, S^j and I^k can similarly be used. We assume a simulator and an oracle to answer to inquired messages. The oracles outputs three states: Accept, reject and $\perp$. If the oracle U^i or S^j is accepted and computes a session key, the following notations are determined; an identity for session (sid_{U^i} *or* sid_{S^j}), an identity for the partner (pid_{U^i} *or* pid_{S^j}) and the session keys (sk_{U^i} *or* sk_{S^j}).

Initialization is done before the simulation. U has the identity ID, password PW and a smart card containing d, e, f, P, q and p. PW is selected of a set with size N. S has parameters c, P, p, q and an identity SID. GW is assigned with an identity ID_{GW} and values x, P, q and p. Moreover, the adversary A knows $ID, SID, ID_{GW}, P, q, p$. In addition, the following definition is used in the simulation:

- **Partnering:** U^i and S^j are partners if a session key is established between them. Beside constructing the session key, four conditions should be satisfied; U^i and S^j are accepted; $sid_{U^i} = sid_{S^j}, pid_{S^j} = U^i, pid_{U^i} = S^j, \cdots, sk_{U^i} = sk_{S^j}$.
- $sfs-fresh$**:** I^k reaches $sfs-fresh$ if the below events are not occurred:
  1) $Reveal(I^k)$
  2) $Reveal(Pid_{I^k})$
  3) Any $Corrupt(I^m)$ query before the $Test$ query, where m is a legitimate participant, containing k.
- $sfs-ake$ *security***:** if A has the advantage on guessing the coin a on P after $Test(I^k)$ where I^k is $sfs-fresh$ and A guesses a bit a', the advantage is defined as

$$Adv_P^{sfs-ake}(A) = 2Pr[a = a'] - 1$$

A scheme is "$sfs-ake$"−secure if $Adv_P^{sfs-ake}(A)$ be a negligible value.

Now, in the form of following theorem, we give the formal proof of our new scheme.

Theorem 1. *The adversary $\mathcal{A}$ can make at most q_s, q_e and q_h queries from Send, Execute and Hash oracles, respectively. $\mathcal{A}$ has the following advantage:*

$$\begin{aligned} Adv_P^{sfs-ake}(A) \leq\ & \frac{(q_s+q_e)^2}{q-1} + \frac{q_h^2+(q_s+q_e)^2}{2^l} \\ & + \frac{12q_h+7q_s}{2^{l-1}} + \frac{2q_s}{N} + 4q_s((q_s+q_e)^2 \\ & + 1)Adv_A^{ECGDH}(t+(2q_s+4q_e)T_s) \end{aligned}$$

Which in the above equation, $\mathcal{P}$ denotes the scheme, G is a cyclic addition group in the field of $E(F_q)$ that has a prime order q and the passwords are chosen from a set with N elements. Additionally, l denotes the length of security parameter. We consider T_m as the needed time for a scalar multiplication in group G.

Proof. The proposed proof of theorem includes of a some related games from the game G_0 to the game G_8. In the test session of the game G_i, the adversary $\mathcal{A}$ guesses the coin a that is denoted by $Succ_i$. Since there is only one user in the proof procedure, there is no need for $\mathcal{A}$ to take time in guessing the user's identity.

- Game G_0: This game simulates the real attacks with random oracles. If one of the following items happens, a random bit like a is selected instead of the answer of $Test$.
  - When the game aborts or stops, $\mathcal{A}$ does not guess.
  - $\mathcal{A}$ makes more queries than the predetermined quantities.
  - $\mathcal{A}$ utilizes more time than the predetermined time.

  In accordance with the upper definition, we have:

$$Adv_P^{sfs-ake}(A) = 2Pr[Succ_0] - 1$$

- Game G_1: In this game, all oracles should be simulated. We also define three lists which the answers to relative queries are stored in them. L_h-list stores the answers to hash queries. If $\mathcal{A}$ asks a hash query, the answer will be stored in L_A-list and the transcripts of all messages are stored in the L_P-list. In order to break the privacy of authentication processes and to obtain the session keys, the adversary $\mathcal{A}$ can make queries to oracles. Then $Pr[Succ_1] = Pr[Succ_0]$ and so, G_0 and G_1 are indistinguishable.

- Game G_2: In this stage, we want to avoid the collisions in the messages. Using the birthday paradox, we introduce the three following collisions:
  - In different sessions, it is possible that the random numbers $\alpha, \beta \in [1, q-1]$ to be used for the same. Note that, in this case, the total probability will be bounded by $\frac{(q_s+q_e)^2}{2(q-1)}$.
  - The three random numbers r_1, r_2 and r_3 may have collisions. The total probibility will be $\frac{(q_s+q_e)^2}{2^{l+1}}$.
  - The upper bound of the probibility of collisions in hash functions is $\frac{q_h^2}{2^{l+1}}$.

    Finally, we can find that $|Pr[Succ_2] - Pr[Succ_1]| \leq \frac{(q_s+q_e)^2}{2(q-1)} + \frac{(q_s+q_e)^2+q_h^2}{2^{l+1}}$.

- Game G_3: During this game, we want to find the probability of forging M_1 without random oracles. Since the simulator $\mathcal{B}$ answers as S, we can add steps to $Send\left(U^i, GW^t, M_1\right)$: the simulator $\mathcal{B}$ needs to check if $M_1 \in L_P - list$ and $(ID \parallel *, *)$, $(* \parallel ID, MI)$, $(* \parallel MI, *)$, $(* \parallel ID, *)$, $(* \parallel B_2, *)$ and $(ID \parallel MI \parallel * \parallel SID, B_5)$ are in L_A-list. If any of these parameters fails, the relative query will be terminated. Since S does not password PW or MI^{new}, $(r_1 \parallel PW, *)$ cannot be exterminated. The probabilities for $(* \parallel ID, MI)$ and $(ID \parallel MI \parallel * \parallel SID, B_5)$ are all bounded by $\frac{q_e}{2^l}$ and other parameters are bounded by $\frac{q_h}{2^l}$. Finally, we can see that $|Pr\left[Succ_3\right] - Pr\left[Succ_2\right]| \leq \frac{(5q_h+2q_s)}{2^l}$.

- Game G_4: In this game, we want to find the probibility of forging M_2 without random oracles. we can add steps to $Send(GW^t, S^j, M_2)$: the simulator $\mathcal{B}$ needs to check if $M_2 \in L_P - list$ and $(SID \parallel *, c)$, $(MI \parallel SID \parallel c \parallel B_2, D_1)$ are in $L_A - list$. The probabilities for $(MI \parallel SID \parallel c \parallel B_2, D_1)$ is bounded by $\frac{q_s}{2^l}$ while for $(SID \parallel *, c)$, this bound is equal to $\frac{q_h}{2^l}$. Therefore, we can see that $|Pr\left[Succ_4\right] - Pr\left[Succ_3\right]| \leq \frac{(q_h+q_s)}{2^l}$.

- Game G_5: During this game, we find the probibility of forging M_3 without random oracles. we can add steps to $Send(GW^t, S^j, M_3)$: the simulator $\mathcal{B}$ needs to check if $M_3 \in L_P - list$ and $(1, B_2 \parallel C_1 \parallel *, *)$, $(MI \parallel SID \parallel * \parallel C_3)$ and $(c \parallel MI \parallel SID \parallel C_4)$ are in $L_A - list$. The probabilities for $(MI \parallel SID \parallel * \parallel C_3)$ and $(c \parallel MI \parallel SID \parallel C_4)$ are bounded by $\frac{q_s}{2^l}$ and for $(1, B_2 \parallel C_1 \parallel *, *)$, this bound is at most equal to $\frac{q_h}{2^l}$. Finally, we can see that $|Pr\left[Succ_5\right] - Pr\left[Succ_4\right]| \leq \frac{(q_h+2q_s)}{2^l}$.

- Game G_6: In this game, we want to find a forge of forging M_4 without random oracles. we can add steps to $Send(GW^t, U^i, M_4)$: the simulator $\mathcal{B}$ requires to verify $M_4 \in L_P - list$ and $(ID_{GW} \parallel * \parallel MI^{new}, *)$, $(MI^{new} \parallel r_3, *)$, $(MI^{new} \parallel *, *)$, $(1, B_2 \parallel C_1 \parallel *, *)$, $(MI \parallel SID \parallel * \parallel C_3)$ and $(ID \parallel MI \parallel MI^{new} \parallel SID \parallel D_2 \parallel D_3 \parallel r_3, *)$ are in $L_A - list$. The last two terms have the upper bound $\frac{q_s}{2^l}$ and the others have at most $\frac{q_h}{2^l}$. So, we can see that $|Pr\left[Succ_6\right] - Pr\left[Succ_5\right]| \leq \frac{(5q_h+2q_s)}{2^l}$.

- Game G_7: In this game, the adversary $\mathcal{A}$ uses random oracles to solve the ECGDH-problem. We modify the h_1 oracle as follows: If $\mathcal{A}$ asks a $(1, \alpha P \parallel \beta P \parallel \lambda)$, the simulator $\mathcal{B}$ checks if $(1, \alpha P \parallel \beta P \parallel *, sk) \in L_A - list$. If there exists such a term, $\mathcal{B}$ returns sk. Otherwise, $\mathcal{B}$ uses the ECDDH oracle to check $\lambda = ?\alpha\beta P$. If this check is failed, $\mathcal{B}$ stops the game and report failure. Otherwise, $\mathcal{B}$ chooses $sk \in \{0,1\}^l$, answers to the query and finally adds $(1, \alpha P \parallel \beta P \parallel \lambda, sk)$ into L_A-list. Here, we intersects the game into two aspects. Firs of all, the adversary $\mathcal{A}$ asks $Corrupt$ $(smart\ card)$-query and then, gets all information of the card.
  - This aspect simulates active attacks. The adversary $\mathcal{A}$ selects a password PW^* with size N. Then, he/she can forge messages to start the session. Since $\mathcal{A}$ can ask at most q_s $Send$-query, the probability of guessing the correct password is $\frac{q_s}{N}$.
  - This aspect simulates passive attacks. Here, we have two cases:
    (a) In orther to break the ECGDH-problem, the adversary $\mathcal{A}$ asks $Execute$-queries and h_1-queries. $\mathcal{A}$ can retrieve from L_A-list with the probability that bounded by $\frac{1}{q_h}$. In this case, the probability is at most $q_h Adv_A^{ECGDH}(t + 4q_e T_m)$.
    (b) In orther to simulate the $Execute$-queries, the adversary $\mathcal{A}$ asks $Send$-queries. Similar to the last case, we can obtain the probability $q_h Adv_A^{ECGDH}(t + 2q_e T_m)$.

    Finally, we have:

$$\begin{aligned} & \left| Pr\left[Succ_6\right] - Pr\left[Succ_5\right] \right| \\ \leq\ & \frac{q_s}{N} + q_h Adv_A^{ECGDH}(t + 4q_e T_m) \\ +\ & q_h Adv_A^{ECGDH}(t + 2q_e T_m) \\ \leq\ & \frac{2q_s}{N} + q_h Adv_A^{ECGDH}(t + (4q_e + 2q_s)T_m) \end{aligned}$$

- Game G_8: This game is about strong forward security. The adversary $\mathcal{A}$ can ask all $Corrupt$-oracles. However, in the light of the $sfs - fresh$ notion, $Corrupt(1^m)$-query should occure after $Test$. So, $\mathcal{A}$ can utilizes the old sessions only. Like game G_7, we can find $(1, \alpha P \parallel \beta P \parallel \alpha\beta P, sk)$ from L_A-list. The probability of obtaining αP and βP in the same session is $\frac{1}{(q_s+q_e)^2}$. Therefore, $|Pr\left[Succ_8\right] - Pr[Succ_7]| \leq 2q_h(q_s + q_e)^2 Adv_A^{ECGDH}(t + (4q_e + 2q_s)T_m)$. This implies that the adversary $\mathcal{A}$ has no more advantage and $Pr[Succ_8] = \frac{1}{2}$.

Finally, Theorem 1 is proved by combining all above games. □

5.3 Formal Verification Using ProVerif

This section analyses the security of the proposed protocol via the ProVerif as one of the most well-known formal automated security analysis tools.

5.3.1 Premises in the Verification

As in [36], first of all, we mention some realties containing: constants, shared keys, channels, equations and functions which are required for analysis of the protocol. The realties are described in Figure 1.
In order to test correspondence relevance for the sensor and the user (during the login and authentication phase), we use four different events. In addition, the first two queries check the session keys security and the last two

verify the correctness of relevances of events. These events are presented in Figure 2.

(*Channels and shared keys are listed below*)
free ch1: channel. (*the public channel between the user and the sensor*)
free ch2: channel. (*the public channel between the sensor and GW*)
free sch1: channel [private]. (*the secret channel between the user and GW*)
free sch2: channel [private]. (*the secret channel between the sensor and GW*)
free sku: bitstring [private]. (*the user's session key*)
free sks: bitstring [private]. (*the sensor's session key*)
(*Constants are listed below*)
free x:bitstring [private]. (*the private key of GW*)
free ID_i:bitstring [private]. (*Ui's identity*)
free PW_i:bitstring [private]. (*Ui's password*)
const $IDGW$:bitstring. (*GW's identity*)
const P:bitstring. (*the generator P*)
const SID_j:bitstring. (*S_j's identity*)
table d(bitstring). (*database in GW*)
(*Functions and equations are listed below:*)
fun h(bitstring):bitstring. (*hash function*)
fun h_1(bitstring):bitstring. (*hash function*)
fun mul(bitstring,bitstring):bitstring.
(*scalar multiplication function*)
fun xor(bitstring,bitstring):bitstring. (*XOR function*)
fun con(bitstring,bitstring):bitstring.
(*string concatenation*)
equation for all m:bitstring, n:bitstring;
$xor(xor(m,n),n)$
$= m$. (*XOR computation*)
equation forall m:bitstring,n:bitstring;
$mul(mul(P,m),n)$
$= mul(mul(P,n),m)$.(*scalar multiplication*)

Figure 1: The ProVerif code definition

Events
event UserStart(bitstring)
event UserAuth(bitstring)
event SensorStart(bitstring)
event SensorAuth(bitstring)
Queries
query attacker(sku)
query attacker(sks)
query id:bitstring; inj-event(UserAuth(id))
==> inj-event(UserStart(id)).
query sid:bitstring; inj-event(SensorAuth(sid))
==> inj-event(SensorStart(sid).

Figure 2: Events and queries in Proverif code

5.3.2 Scheme Model

We simulate our proposed scheme in parallel execution steps. Moreover, there are three entities in our scheme as participants and each participant has its own process:

$$process!User|!GW|!Sensor.$$

The processes of the user, the sensor and the gateway are mentioned in Figure 3, Figure 4 and Figure 5, respectively. The processes of the user and the sensor can be divided into two separated parts: registration and authentication. The process of the gateway includes three parts: two parts for registration and one part for authentication.

5.3.3 The Verification Results

The final main results are shown in Figure 6. It determines that the session keys are secure via the verification.

5.3.4 Comparison

In this section, we compare our proposed scheme with other schemes from both of the security and performance points of views. We want to compare our proposed scheme with some recent well-known schemes: Wu *et al.*'s scheme ([36]), Hsieh *et al.*'s scheme ([14]), Shi *et al.*'s scheme ([31]) Choi *et al.*'s scheme ([6]), Chang *et al.*'s scheme ([4]) and Farash *et al.*'s scheme ([10]).

Please note that since there are two versions of Chang *et al.*'s scheme ([4]): One is based on the hash functions and the other one is based on the elliptic curve cryptography, we use S1 and S2 to denote the versions.

Security comparison:

Although Wu *et al.* claimed that their proposed scheme is resistant against to replay attack and user forgery attack, however we showed that their scheme is vulnerable against these attacks.

In the security comparison posed in Table 4, we consider these security properties: Insider attack, off-line guessing attack, user forgery attack, gateway forgery attack, sensor capture attack, desyncronization attack, replay attack, known-key attack, user anonymity and strong forward security.

Performance comparison:

In this section, we discuss about performance of our scheme and compare it with some related schemes. Table 5 presents the comparison and uses the following notations and considerations:

- T_s denotes the time cost of a scalar multiplication in G and T_h is the time for a hash computation. In accordance with the Xu *et al.*'s scheme ([37]), we can see that $T_s \gg T_h$.

Table 4: Comparison of the security parameters

	Our scheme	[36]	[14]	[32]	[31]	[6]	[4] (S1)	[4] (S2)	[10]
Immune to the insider attack	✓	✓	×	✓	✓	✓	✓	✓	✓
Immune to the off-line guessing attack	✓	✓	×	×	×	×	×	×	×
Immune to the user forgery attack	✓	×	×	×	×	×	✓	✓	✓
Immune to the gateway forgery attack	✓	✓	✓	✓	✓	✓	✓	✓	✓
Immune to the sensor capture attack	✓	✓	×	×	✓	✓	✓	✓	✓
Immune to the de-syncronization attack	✓	✓	✓	✓	✓	✓	✓	✓	✓
Immune to the replay attack	✓	×	✓	✓	✓	✓	✓	✓	✓
Immune to the known-key attack	✓	✓	×	✓	✓	✓	✓	✓	✓
User anonymity	✓	✓	✓	×	×	×	✓	✓	✓
Strong forward security	✓	✓	×	×	✓	✓	×	✓	✓

- We consider that the points in G has totally 320 bits. The security parameter l is 160-bit and hence, the length of secret parameters such as x in the gateway, random numbers, the hash results and SID_j are 160-bits. Moreover, we use Q_u and Q_s to denote the quantities of the users and the sensors in the WSN. $|P|$, $|p|$ and $|q|$ are lengths for the parameters P, p and q such that $|p| \approx 160$ and $|q| \approx 160$.
- In Table 5, we show $(Q_u + Q_s + 1)$ with the Q_T.

6 Conclusion

In this paper, we firstly discussed on the security evaluation of the Wu *et al.*'s user authentication scheme and showed that their scheme is vulnerable against forgery attack and DoS attack. After that, in order to eliminate the weaknesses, we proposed an improved user authentication scheme. In addition, we presented a formal security analysis of our scheme via ProVerif and we suggested a provable security for the proposed scheme. Finally, we compared security and efficiency of our proposed scheme with some related schemes which indicate that the proposed scheme is a well-performed, secure and more practical scheme for IoT communications.

References

[1] A. Akbarzadeh, M. Bayat, B. Zahednejad, A. Payandeh, and M. R. Aref, "A lightweight hierarchical authentication scheme for internet of things," *Journal of Ambient Intelligence and Humanized Computing*, July 2018. (https://doi.org/10.1007/s12652-018-0937-6)

[2] A. A. Alamr, F. Kausar, J. Kim, and C. Seo, "A secure ECC-based RFID mutual authentication protocol for internet of things," *The Journal of Supercomputing*, vol. 74, no. 9, pp. 4281–4294, 2018.

[3] E. Bresson, O. Chevassut, and D. Pointcheval, "Security proofs for an efficient password-based key exchange," in *Proceedings of the 10th ACM Conference on Computer and Communications Security*, pp. 241–250, 2003.

[4] C. C. Chang and H. D. Le, "A provably secure, efficient, and flexible authentication scheme for ad hoc wireless sensor networks," *IEEE Transactions on Wireless Communications*, vol. 15, no. 1, pp. 357–366, 2016.

[5] T. H. Chen and W. K. Shih, "A robust mutual authentication protocol for wireless sensor networks," *ETRI Journal*, vol. 32, no. 5, pp. 704–712, 2010.

[6] Y. Choi, D. Lee, J. Kim, J. Jung, J. Nam, and D. Won, "Security enhanced user authentication protocol for wireless sensor networks using elliptic curves cryptography," *Sensors*, vol. 14, no. 6, pp. 10081-106, 2014.

[7] M. L. Das, "Two-factor user authentication in wireless sensor networks," *IEEE Transactions on Wireless Communications*, vol. 8, no. 3, pp. 1086–1090, 2009.

[8] P. K. Dhillon and S. Kalra, "Secure multi-factor remote user authentication scheme for internet of things environments," *International Journal of Communication Systems*, vol. 30, no. 16, pp. e3323, 2017.

[9] R. Fantacci, T. Pecorella, R. Viti, and C. Carlini, "A network architecture solution for efficient iot wsn backhauling: challenges and opportunities," *IEEE Wireless Communications*, vol. 21, no. 4, pp. 113–119, 2014.

[10] M. S. Farash, M. Turkanović, S. Kumari, and M. Hölbl, "An efficient user authentication and key agreement scheme for heterogeneous wireless sensor network tailored for the internet of things environment," *Ad Hoc Networks*, vol. 36, pp. 152–176, 2016.

[11] X. Feng, X. Liu, and H. Yu, "A new internet of things group search optimizer," *International Journal of Communication Systems*, vol. 29, no. 3, pp. 535–552, 2016.

[12] H. Hayouni, M. Hamdi, and T. H. Kim, "A survey on encryption schemes in wireless sensor networks," in *7th International Conference on Advanced Software Engineering and Its Applications (ASEA'14)*, pp. 39–43, 2014.

[13] D. He, Y. Gao, S. Chan, C. Chen, and J. Bu, "An enhanced two-factor user authentication scheme in

h

Table 5: Comparison of performance

	our scheme	[36]	[14]	[32]	[31]	[6]	[4] (S1)	[4] (S2)	[10]
User's complexity	$2T_m + 13T$	$2T_m + 13T_h$	$8T_h$	$7T_h$	$3T_m + 5T_h$	$3T_m + 7T_h$	$7T_h$	$2T_m + 7T_h$	$11T_h$
Sensor's complexity	$2T_m + 4T_h$	$2T_m + 4T_h$	$2T_h$	$5T_h$	$2T_m + 4T_h$	$2T_m + 4T_h$	$5T_h$	$2T_m + 5T_h$	$7T_h$
Gateway's complexity	$1T_m + 13T_h$	$13T_h$	$5T_h$	$7T_h$	$T_m + 4T_h$	$T_m + 4T_h$	$8T_h$	$9T_h$	$14T_h$
Communication Cost (bits)	3680	3680	1280	4000	3840	4220	2720	3040	3520
Private number stored in the Gateway(bits)	160	160	160	$160Q_T$	320	320	$160Q_T$	$160Q_T$	160
Security for IoT	✓	×	×	×	×	×	×	×	×

wireless sensor networks." *Ad Hoc & Sensor Wireless Networks*, vol. 10, no. 4, pp. 361–371, 2010.

[14] W. B. Hsieh and J. S. Leu, "A robust user authentication scheme using dynamic identity in wireless sensor networks," *Wireless Personal Communications*, vol. 77, no. 2, pp. 979–989, July 2014.

[15] M. S. Hwang, Li-Hua Li, "A New Remote User Authentication Scheme Using Smart Cards", *IEEE Transactions on Consumer Electronics*, vol. 46, no. 1, pp. 28–30, Feb. 2000.

[16] R. Kantola, H. Kabir, and P. Loiseau, "Cooperation and end-to-end in the internet," *International Journal of Communication Systems*, vol. 30, no. 12, pp. e3268, 2017.

[17] M. K. Khan and K. Alghathbar, "Cryptanalysis and security improvements of 'two-factor user authentication in wireless sensor networks'," *Sensors*, vol. 10, no. 3, pp. 2450–2459, 2010.

[18] P. Kumar and H. J. Lee, "Cryptanalysis on two user authentication protocols using smart card for wireless sensor networks," in *Wireless Advanced (WiAd'11)*, pp. 241–245, 2011.

[19] S. Kumari, X. Li, F. Wu, A. K. Das, K. K. R. Choo, and J. Shen, "Design of a provably secure biometrics-based multi-cloud-server authentication scheme," *Future Generation Computer Systems*, vol. 68, pp. 320–330, 2017.

[20] S. Kumari, X. Li, F. Wu, A. K. Das, H. Arshad, and M. K. Khan, "A user friendly mutual authentication and key agreement scheme for wireless sensor networks using chaotic maps," *Future Generation Computer Systems*, vol. 63, pp. 56–75, 2016.

[21] S. Kumari, "Design flaws of "an anonymous two-factor authenticated key agreement scheme for session initiation protocol using elliptic curve cryptography," *Multimedia Tools and Applications*, vol. 76, no. 11, pp. 581–583, June 2017.

[22] S. Kumari, M. Karuppiah, A. K. Das, X. Li, F. Wu, and N. Kumar, "A secure authentication scheme based on elliptic curve cryptography for iot and cloud servers," *The Journal of Supercomputing*, pp. 1–26, 2017.

[23] C. T. Li, M. S. Hwang, "A lightweight anonymous routing protocol without public key en/decryptions for wireless ad hoc networks",*Information Sciences*, vol. 181, no. 23, pp. 5333–5347, Dec. 2011.

[24] C. T. Li, M. S. Hwang and Y. P. Chu, "An efficient sensor-to-sensor authenticated path-key establishment scheme for secure communications in wireless sensor networks", *International Journal of Innovative Computing, Information and Control*, vol. 5, no. 8, pp. 2107-2124, Aug. 2009.

[25] W. T. Li, T. H. Feng, and M. S. Hwang, "Distributed detecting node replication attacks in wireless sensor networks: A survey," *International Journal of Network Security*, vol. 16, no. 5, pp. 323–330, 2014.

[26] Z. Liu, E. Wenger, and J. Großschädl, "Mote-ecc: Energy-scalable elliptic curve cryptography for wireless sensor networks," in *International Conference on Applied Cryptography and Network Security*, pp. 361–379, 2014.

[27] P. Middleton, P. Kjeldsen, and J. Tully, "Forecast: The internet of things, worldwide," *Gartner Research*, 2013. (https://www.gartner.com/doc/2625419/forecast-internet-things-worldwide-)

[28] K. T. Nguyena, M. Laurentb, N. Oualha,"Survey on secure communication protocols for the internet of things," *Ad Hoc Networks*, vol. 32, pp. 17–31, 2015.

[29] S. Rostampour, N. Bagheri, M. Hosseinzadeh, and A. Khademzadeh, "A scalable and lightweight grouping proof protocol for internet of things applications," *The Journal of Supercomputing*, vol. 74, no. 1, pp. 71–86, 2018.

[30] O. Said, "Analysis, design and simulation of internet of things routing algorithm based on ant colony optimization," *International Journal of Communication Systems*, vol. 30, no. 8, pp. e3174, 2017.

[31] W. Shi and P. Gong, "A new user authentication protocol for wireless sensor networks using elliptic curves cryptography," *International Journal of Distributed Sensor Networks*, vol. 9, no. 4, pp. 730831, 2013.

[32] M. Turkanović, B. Brumen, and M. Hölbl, "A novel user authentication and key agreement scheme for heterogeneous ad hoc wireless sensor networks, based on the internet of things notion," *Ad Hoc Networks*, vol. 20, pp. 96–112, 2014.

[33] B. Vaidya, D. Makrakis, and H. T. Mouftah, "Improved two-factor user authentication in wireless sensor networks," in *IEEE 6th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob'10)*, pp. 600–606, 2010.

```
The user's process:
let User=
new r0:bitstring;
let MPi=h(con(r0,PWi)) in
let MIi=h(con(r0,IDi)) in
out(sch1,(MPi,MIi,IDi));
in(sch1,(xei:bitstring,xfi:bitstring));
let ei = xei in
let fi = xfi in
let di = xor(h(con(IDi,PWi)),r0) in
!
(
event UserStart(IDi);
let r1 = xor(di,h(con(IDi,PWi))) in
let MIi' = h(con(r1,IDi)) in
let MPi' = h(con(r1,PWi)) in
new alpha:bitstring;
new r2:bitstring;
new r3:bitstring;
new ti':bitstring;
let MIinew = h(con(r2,IDi)) in
let B1= xor(xor(ei,MPi'),r3) in
let B2 = mul(P,alpha) in
let B3 = xor(xor(xor(fi,MIi'),MIinew),
h(con(r3,MIi'))) in
let B4 = xor(IDi,h(con(con(r3,MIinew),B2))) in
let B5 = h(con(con(con(IDi,MIi'),MIinew),SIDj)) in
let M1 =(MIi',SIDj,B1,B2,B3,B4,B5) in
out(ch1,M1);
in (ch1,(xC1:bitstring,xC3:bitstring,xD2:bitstring,
xD3:bitstring,xD4:bitstring));
if xD4 = h(con(con(con(con(con(con(IDi,MIi'),
MIinew),SIDj),xD2),xD3),r3)) then
let B6 = mul(xC1,alpha) in
let sku = h1(con(con(B2,xC1),B6)) in
if xC3 = h(con(con(MIi',SIDj),sku)) then
let dinew = xor(r2,h(con(IDi,PWi))) in
let einew = xor(xor(xD2,h(con(MIinew,r3))),
h(con(r2,PWi))) in
let finew = xor(xor(xD3,MIinew),h(con(MIi',r3))) in
let di = dinew in
let ei = einew in
let fi = finew in
0).
```

Figure 3: Code for the user's role

```
The sensor's process:
let Sensor =
out(sch2,SIDj);
in(sch2, xxcj:bitstring);
!
(
in(ch2,(uMIi:bitstring,uSIDj:bitstring,uB2:bitstring,
    uD1:bitstring, xxC0:bitstring));
if uSIDj = SIDj then
if uD1 = h(con(con(con(uMIi,uSIDj),xxcj),uB2)) then
event SensorStart(uSIDj);
new beta:bitstring;
let C1 = mul(P,beta) in
let C2 = mul(uB2,beta) in
let sks = h1(con(con(uB2,C1),C2)) in
let C3 = h(con(con(uMIi,SIDj),sks)) in
let C4 = h(con(con(con(mul(xxcj,xxC0),uMIi),SIDj),Yj)) in
let M3 = (C1,C3,C4) in
out(ch2,M3);
0
).
```

Figure 4: Code for the sensor's role

```
User registration
let GWReg1 =
in(sch1,(xMPi:bitstring,xMIi:bitstring,xIDi:bitstring));
let ei'= xor(con(con(IDGW,x),xMIi),xMPi) in
let fi'= xor(h(con(xMIi,x)),xMIi) in
insert d(xIDi);
out (sch1,(ei',fi')).
Sensor registration
let GWReg2 =
in(sch2,(ySIDj:bitstring));
let cj = h(con(ySIDj,x)) in
out(sch2,(cj)).
Authentication
let GWAuth =
in(ch1,(xxMIi:bitstring,xxSIDj:bitstring,xxB1:bitstring,
xxB2:bitstring, xxB3:bitstring,xxB4:bitstring,xxB5:bitstring));
let xr3 = xor(xxB1,con(con(IDGW,x),xxMIi)) in
let xMIinew = xor(xor(xxB3,h(con(xxMIi,x))),
h(con(xr3,xxMIi))) in
let xIDi = xor(xxB4,h(con(con(xr3,xMIinew),xxB2))) in
get d(=xIDi) in
new lambda:bitstring;
let C0 = mul(P,lambda) in
if xxB5 = h(con(con(con(xIDi,xxMIi),xMIinew),xxSIDj)) then
event UserAuth(xIDi);
let pcj = h(con(xxSIDj,x)) in
let xxD1 = h(con(con(con(xxMIi,xxSIDj),mul(pcj,C0)),xxB2)) in
let M2 =(xxMIi,xxSIDj,xxB2,xxD1,C0) in
out (ch2,M2);
in (ch2,(xxC0:bitstring,xxC1:bitstring,xxC3:bitstring,xxC4:bitstring));
if xxC4 = h(con(con(mul(pcj,C0),xxMIi),xxSIDj)) then
event SensorAuth(xxSIDj);
let D2 = xor(h(con(con(IDGW,x),xMIinew)),
h(con(xMIinew,xr3))) in
let D3 = xor(h(con(xMIinew,x)), h(con(xxMIi,xr3))) in
let D4 =
con(con(con(con(con(con(xIDi,xxMIi),xMIinew),xxSIDj),
D2),D3),xr3) in
let M4 = (xxC1,xxC3,D3,D4,D5) in
out(ch1,M4).
```

Figure 5: Code for the gateway's role

Figure 6: Results of the verification by ProVerif

[34] R. Watro, D. Kong, S. f. Cuti, C. Gardiner, C. Lynn, and P. Kruus, "Tinypk: Securing sensor networks with public key technology," in *Proceedings of the 2nd ACM Workshop on Security of Ad Hoc and Sensor Networks*, pp. 59–64, 2004.

[35] F. Wu, L. Xu, S. Kumari, and X. Li, "A new and secure authentication scheme for wireless sensor networks with formal proof," *Peer-to-Peer Networking and Applications*, vol. 10, no. 1, pp. 16–30, 2017.

[36] F. Wu, L. Xu, S. Kumari, and X. Li, "A privacy-preserving and provable user authentication scheme for wireless sensor networks based on internet of things security," *Journal of Ambient Intelligence and Humanized Computing*, vol. 8, no. 1, pp. 101–116, Feb. 2017.

[37] L. Xu and F. Wu, "Cryptanalysis and improvement of a user authentication scheme preserving uniqueness and anonymity for connected health care," *Journal of Medical Systems*, vol. 39, no. 2, p. 10, Jan. 2015.

[38] H. L. Yeh, T. H. Chen, P. C. Liu, T. H. Kim, and H. W. Wei, "A secured authentication protocol for wireless sensor networks using elliptic curves cryptography," *Sensors*, vol. 11, no. 5, pp. 4767–4779, 2011.

[39] H. L. Yeh, T. H. Chen, P. C. Liu , T. H. Kim and H. W. Wei, "A secured authentication protocol for wireless sensor networks using elliptic curves cryptography," *Sensors*, vol. 11, no. 5, pp. 4767–4779, 2011.

[40] Y. Y. Zhang, X. Z. Li, and Y. A. Liu, "The detection and defence of DoS attack for wireless sensor network," *The Journal of China Universities of Posts and Telecommunications*, vol. 19, pp. 52–56, 2012.

Biography

Majid Bayat received his Ph.D. from the Department of Mathematics and Computer Sciences at Kharzmi University in Tehran, Iran. He is presently an assistant professor of Computer Engineering of Shahed University, Tehran, Iran. His research interests include cryptographic protocols, smart grid and IoT security.

Mohammad Beheshti Atashgah is a PhD candidate at ICT Complex, Malek-Ashtar University of Technology Tehran, Iran. He is a researcher in Information System and security lab (ISSl) in Sharif University. His research interests include IoT security and provable security.

Morteza Barari was born in Freydoonkenar, Iran. He received a Ph.D. degree from AmirKabir University of Technology in 2003. He is currently a faculty member at the Department of Electrical Engineering of the Malek-Ashtar University of Technology, Tehran, Iran. He has published more than 50 papers and 2 books. His research interests are in stochastic signal processing, radar design, satellite communication, and adaptive array processing.

Mohammad Reza Aref received the B.Sc. degree in 1975 from the University of Tehran, Iran, and the M.Sc. and Ph.D. degrees in 1976 and 1980, respectively, from Stanford University, Stanford, CA, USA, all in electrical engineering. He returned to Iran in 1980 and was actively engaged in academic affairs. He was a Faculty member of Isfahan University of Technology from 1982 to 1995. He has been a Professor of electrical engineering at Sharif University of Technology, Tehran, since 1995, and has published more than 230 technical papers in communication and information theory and cryptography in international journals and conferences proceedings. His current research interests include areas of communication theory, information theory, and cryptography.

Identity Management Security Authentication Based on Blockchain Technologies

Pengfei Fan[1], Yazhen Liu[1], Jiyang Zhu[1], Xiongfei Fan[2], and Liping Wen[3]
(Corresponding author: Pengfei Fan)

Information Communication Operation and Maintenance Center,
Information and Communication Branch, State Grid Inner Mongolia Eastern Electric Power Co., Ltd.[1]
Training Center, Inner Mongolia Power (Group) Co., Ltd.[2]
Hohhot Power Supply Bureau, Inner Mongolia Power (Group) Co., Ltd.[3]
Hohhot, Inner Mongolia Autonomous Region 010020, China
(Email: lyzfpf@126.com)
(Received Sept. 22, 2018; Revised and Accepted June 12, 2019; First Online Oct. 1, 2019)

Abstract

In recent years, the popularity of the Internet and computers has made people's communication more convenient and faster, and access to information has become faster and faster. However, due to the openness of the Internet, how to ensure the legal and credible identity in the communication process has become an important part of Internet security. This study briefly introduced the block chain and the block chain-based identity security authentication system and simulated and analyzed the block chain and security of the system. The results showed that the block chain could accurately authenticate the user identity information of the input public key after issuing valid digital certificate and prevent the non-authenticated identity information user from viewing the digital certificate. The increase of block chain nodes also increased the fault-tolerant nodes. At the same time, the endorsement conditions which were that the normal nodes were larger than half of all the nodes made the nodes of the system used safely as long as the number of damaged nodes was no more than half. The hackers needed to successfully attack more than half of the nodes before tampering the system data, but this operation was almost impossible to achieve objectively. Thus, the security of the system was extremely high.

Keywords: Block Chain; Digital Certificate; Identity Authentication; Smart Contract

1 Introduction

In recent years, the popularity of the Internet and computers has made people's lives more convenient [13]. The most intuitive one is the exchange and acquisition of information. However, the emergence of the Internet has not only brought convenience, but also brought information security risks. The Internet has openness and anonymity [7]. The former is an important factor in the development of the Internet. The latter has formed a hidden danger after combining the former. Users participating in the Internet cannot guarantee whether the communication object is trustworthy. Therefore, identity management has become one of the important technologies for Internet information security.

The basic principle of the identity authentication management technology [9] is to generate a unique digital certificate for the application user as the identity certificate. At the beginning, limited by technology, the identity management system is mainly a centralized system [11], which is that digital certificates and keys are provided by third parties, and identity information is also kept by third parties. The centralized identity authentication system protects the user's identity information to some extent, but it has obvious shortcomings and cannot guarantee the credibility of the third party, including termination of third-party service, data loss or malicious leakage of data. People's demand for identity authentication systems is not met until the emergence of block chain technology.

The important features of block chain technology [4] are decentralization and collective maintenance. The former is the same authority between nodes, while the latter is that all nodes share the identity information authentication, and the authentication process is transparent, open and credible. Yu *et al.* [14] proposed an effective social network information privacy protection algorithm, which used the block chain to store the user's public key and encrypted the plaintext by hybrid hash encryption algorithm after binding. The simulation results showed that the algorithm could effectively defend against different types of attacks. Lin *et al.* [8] proposed a block chain-based secure mutual authentication system, BSEIN, to implement an access control policy. The system could provide privacy protection such as anonymous authenti-

cation, and the system had good scalability due to the smart contract of the block chain.

The performance evaluation results showed that the system's response speed was excellent. Guan *et al.* [15] divided users into different groups. Each group has a private block chain to record the data of its members and use pseudonyms to protect user privacy with Bloom filter for fast authentication. The experimental analysis showed that the method could meet the security requirements and the performance was better than other common methods. This study briefly introduced the block chain and the block chain-based identity security authentication system and simulated and analyzed the block chain and security of the system.

2 Blockchain

As shown in Figure 1, block chain has six block tables. The blocks from bottom to top are linked in chronological order on a cryptographic basis. Block chain technology adopts timestamp proofing, cryptography and other technologies, coupled with the distributed storage structure of the block to make the block chain decentralized and difficult to falsify forgery and collective maintenance, which ensures the security and privacy of important data in the block. At the same time, the block chain can also be regarded as a state machine that constantly changes its state through transactions. Its evolution formula [12] is:

$$\theta_{t+1} \equiv Y(\theta_t, T),$$

where θ_t represents the block chain state at time t, T is a transaction, and $Y(\cdot)$ is a state transition function. After the transaction has evolved for a period of time, the verified transaction is collected into the block, and the block is connected by hash value. The state conversion formula [2] is:

$$\begin{aligned} \theta_{t+1} &\equiv \prod(\theta_t, B) \\ B &\equiv ((T_0, T_1, \cdots), \cdots), \end{aligned}$$

where $\prod(\cdot)$ is a block B-based transaction conversion function, and the block B contains transactions T and other data.

In the view of structure, the data layer is the lowest layer, and the block encapsulates the basic unit as transaction data and uses cryptography such as Hash algorithm and encryption algorithm to construct the linked data in chronological order. The encryption algorithm is divided into two types: symmetric and asymmetric. The former encrypts and decrypts with one key, while the latter is divided into private key and public key. The derivation between the two is irreversible.

The network layer is the main manifestation of the decentralization of block chain. The content of its package includes network architecture, inter-block communication protocol and authentication method [5]. After a long period of development, the block chain usually adopts a point-to-point (P2P) network architecture. In this network architecture, the computer nodes participating in it provide the same service through the topology, so there is no central service in the block chain. The consensus layer encapsulates all the consensus mechanism algorithms between the nodes in the block chain. Due to the decentralization of the block chain, the "books" of each node are highly dispersed, thus, a consensus algorithm is needed to select the most suitable node to perform "billing rights." The process of running a consensus algorithm to select a node is called "mining." The commonly used consensus algorithms are: workload proof, equity certificate, entrusted equity certificate, *etc.* This study adopted the most secure consensus algorithm, workload proof [10], which is currently recognized.

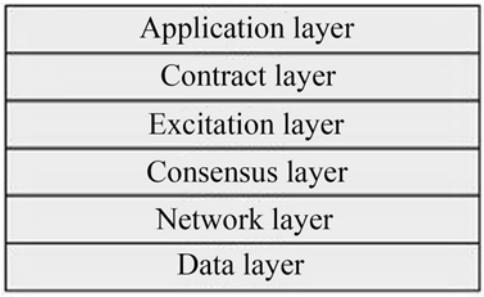


Figure 1: Framework of blockchain

The above data layer, network layer and consensus layer are the necessary and indispensable factors of the block chain. In addition, the incentive layer is used to reward the structure of the nodes involved in the "mining" of the block chain. It stimulates a large number of nodes to participate in "mining" through rewards, thereby realizing the stability and security of the block chain by means of consensus mechanism. The contract layer encapsulates a piece of contract code that is executed when the pre-defined conditions are met.

3 Blockchain-based Identity Authentication System

3.1 Overall Structure

As shown in Figure 2, the overall architecture of the block chain-based identity authentication system [1] is divided into three parts: an identity authentication system with primary functions, a third-party publicity module for inquiring, and block chain module for privacy security connected with two modules. The identity authentication system is a functional manifestation of the entire system, including a registration system, a certificate issuance system, a block chain management, and an SDK, and the registration system implements a traditional user identity

registration function.

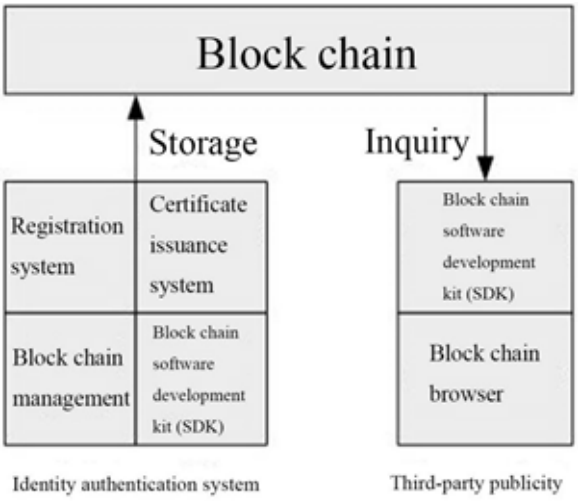


Figure 2: The overall structure of the identity authentication system based on the block chain security system

The certificate issuance system implements the traditional user identity authentication function, but unlike the traditional one, the module only has the function of issuing a certificate, and the specific digital certificate and related operation records are performed in the block chain. The main function of the block chain management module is to manage the nodes in the block chain, the consensus policy, the smart contract, *etc.*, and the authority for the management operation is only owned by the corresponding administrator, and the execution of the operation requires the consent of multiple parties to pass. The block chain SDK is responsible for connecting the authentication system and block chain, storing identity data operations into the block chain, and receiving information from the block chain.

The third-party publicity module is a module for system users, which includes a block chain SDK and a browser. The role of the SDK is similar to that of the SDK of the authentication system. The public module is connected to the block chain, and the operation information is input and the authentication information is inquired. A browser is a third-party platform that displays or inquires the authentication process, mainly referring to web pages.

The block chain is an important module for the privacy security of the system. This study used the callback function to form the smart contract [6]. The smart contract includes logical operations such as certificate storage and query, issuing key loss management, and certificate revocation management. The execution of a complete smart contract includes three steps of node signature, consensus calculation and accounting. At the same time, in order to improve the credibility of the smart contract call data, the contract can be executed when the node signature is not less than three.

3.2 Certificate Management

For the identity authentication system, the management of the certificate is a crucial part. The issuance, replacement and revocation of the certificate are related to the generation, change and cancellation of the user's identity rights in the system [3]. The most important certificate issuance process is shown in Figure 3. First, the user sends a certificate request to the registration center of the system. The content of the application includes the user digital certificate type, public key and public key validity period, and the information is unique. After receiving the application information, the registration center will automatically or manually review the information and send the certification license to the issuing center after the approval.

After receiving the certification, the issuing center generates a digital certificate according to the template. After receiving the certification, the issuing center generates a digital certificate according to the template. The certificate contents include the serial number and signature algorithm for proving the validity, the holder information for proving the ownership, the public key and validity period for protecting the privacy, and the information of issuer for proving the source. After the certificate is generated, the smart contract is called to verify it. After the signature algorithm is passed, the certificate is stored in the block chain to ensure that the identity corresponding to the digital certificate is transparent and cannot be falsified. After the block chain is successfully deposited, the information of successful operation will be fed back step by step, and the user will be notified by mail or telephone. Compared to traditional identity authentication systems, block chain-based authentication systems are initiated by the user side in generating keys for privacy protection. At the same time, the digital certificate generated by the formal route and stored in the block chain can be inquired through the block chain. When the inquiry cannot be operated, the certificate has expired or is leaked, and the certificate update or revocation is required, and the certificate issuance process is similar.

4 System Performance Test

4.1 Experimental Environment

As shown in Figure 4, the experiment was carried out on the primary server of the lab. The parameters of the primary server were quad-core i7CPU, 16G memory and 1024G hard disk. The Virtualbox software was used to divide two virtual machines (VMs) in the server. The configuration parameters were dual-core CPU, 2G memory, 40G hard disk, acting as the identity authentication system center. VM2 configuration parameters were dual-core CPU, 3G memory, 40G hard disk. 360 browser was used in the paimary server for registering and querying digital certificates in the authentication system of VM1, and VM1 and VM2 were connected by a block chain SDK. To

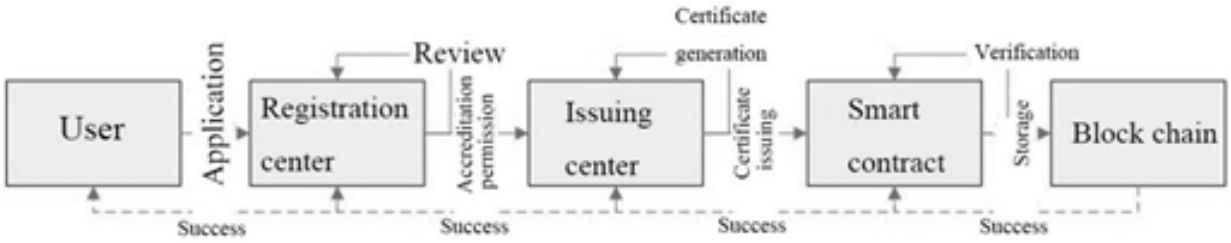


Figure 3: Process of certificate management

facilitate the simulation, the parameters of the nodes in the block chain network were uniformly set as single-core i5CPU, 2.5 GHz working frequency, and 4 G memory.

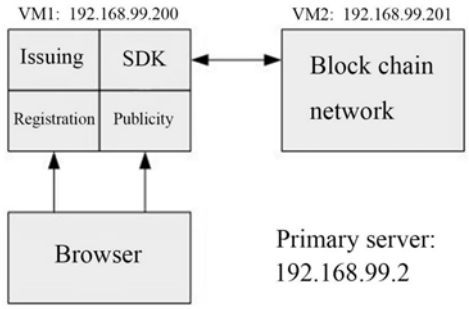


Figure 4: Structure diagram of system test based on blockchain

4.2 Test Content

4.2.1 Identity Authentication Test

Firstly, the user private key was generated by using the RSA algorithm in the Openssl tool. Then, the user public key was generated according to the private key, and the certificate was applied to the browser registration interface by using the public key. After the necessary identity information was successfully applied, the Openssl tool was used to randomly generate another private key and public key. Then, the two public keys were respectively applied through the block chain SDK to simultaneously inquire the three nodes in the block chain for the previously applied digital certificate, and the inquiry result was recorded.

4.2.2 System Security Test

First, three nodes were set for block chain, and two of them were verified, i.e., the operation could be performed when the node signature was not less than two. After that, a certificate was issued, and its validity was checked. When the certificate was invalid or not found, the identity information was leaked. One of the nodes was stopped to simulate the node being hacked, and the rest of the nodes worked normally as issuing or updating the certificate and inquiring the validity. Then, after another node was stopped, the operation of issuing or updating the certificate and inquiring the validity continued. When it was invalid, the next step was to resume the node work one by one and inquire the validity separately. The number of nodes was gradually increased in the block chain, and the number of nodes passing through was always kept larger than half of all nodes. For each additional number of nodes in the block chain, the previous steps were repeated to test the number of fault-tolerant nodes under different node numbers of the system.

4.3 Test Results

4.3.1 Test Results of Identity Authentication

Due to space limitations, only the public key used for the certificate application was listed. As shown in Figure 5, the public key had a length of 1024 bits.

As shown in Table 1, the three nodes with the correct public key for the digital certificate could pass the identity authentication. The random public key was used for the digital certificate inquiry, all the three nodes could not pass the identity authentication, and the interface showed that “There is no such certificate. Please inquire if your information is entered correctly.”, after it returned. It could be seen that the block chain could effectively authenticate the identity information of the valid digital certificate and prevent the non-authenticated identity information user from viewing the digital certificate.

4.3.2 Test Results of System Security

As shown in Figure 6, as the number of nodes involved in ”mining” in the block chain increased, the passing conditions of the system were constantly adjusted, and the conditions of more than half of all nodes were always maintained, and the number of fault-tolerant nodes was also rising. For example, when there were ten nodes in the block chain, the passing condition was six nodes, and the fault-tolerant node was four. This meant that even if four nodes in the block chain were abnormal due to hacking, a bad inquiry signature was issued, or the work was stopped. The entire system could still issue, update and revoke identity certificates in a normal and safe way. At

```
root@coco-Vritualbox:~# openssl rsa -in rsa_private_key.pem -pubout -out
rsa_public_key.pem
Writing RSA key
root@coco-Vritualbox:~# more rsa_public_key.pem
------BEGIN PUBLIC KEY------
Jfaifofazuiofhqwiof Hfjiwfjia U83rjaoif94ioaj9q
fowfu4ifaH&U)UUOOUjoij8908t65nUOIJHIio
Aljiafjifjf824r943090
------END PUBLIC KEY------
```

Figure 5: Public key for certificate application

Table 1: Inquiry results of two public keys

Operations	Inquiry result of Node 1	Inquiry result of Node 2	Inquiry result Node 3
Apply correct public key to inquire	Certification passed	Certification passed	Certification passed
Apply random public key to inquire	Certification failed	Certification failed	Certification failed

the same time, the number of fault-tolerant nodes in the process of restoring the stopped nodes in the experiment process did not change. The reason was that the recovery of the nodes was equivalent to the consensus mechanism of the newly joined nodes participating in the competition for "billing rights". This process was equivalent to automatic synchronization data, which was a feature of the block chain that maintained usability.

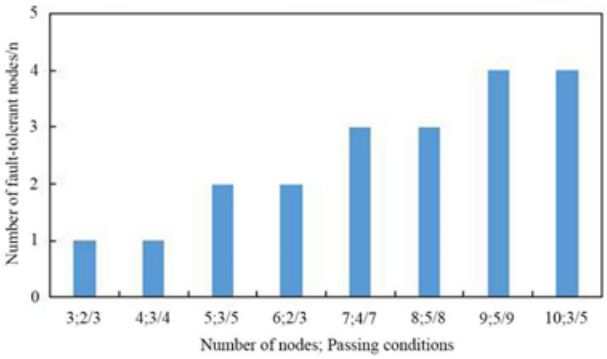


Figure 6: Number of fault-tolerant nodes under different nodes and passing conditions

The passing condition of the block chain was that the passing nodes was generally larger than half of all the nodes. At the same time, if the hacker wanted to destroy, steal or tamper with the digital certificate in the block chain, more than half of the nodes in the chain were needed to crack and attack since the nodes in the block chain were the same in the authority status. The encryption algorithm of the single node of the system required multiple computers to solve and the extremely long time simultaneously. As the nodes in the block chain increased, the fault-tolerant nodes also increased, and the nodes that needed to be cracked also increased, resulting in that it was almost impossible to solve more than half of all the nodes in the block chain at the same time in the actual implementation. Thus, block chain-based identity security authentication systems were extremely secure.

5 Conclusion

This study briefly introduced the block chain and the block chain-based identity security authentication system and simulated and analyzed the block chain and security of the system. By setting different test scenarios of the block chain, the block time and TPS were generated to measure the performance of the block chain. The security of the system were analyzed by stopping and restoring the node work to simulate hacker attacks. The results are that after the block chain issues a valid certificate according to the public key, it can accurately authenticate the user identity information of the input public key and prevent the non-authenticated identity information user from viewing the digital certificate. With the increase of nodes in the block chain, the passing conditions are continuously adjusted under the premise of guaranteeing that the passing nodes are more than half of all the nodes, and the fault-tolerant nodes are increased. The restored nodes can participate in the consensus mechanism normally. The difficulty of hacking attacks on tampering certificates is increasing. Security increases dramatically as nodes increase.

References

[1] M. Benchoufi, R. Porcher, P. Ravaud, "Blockchain protocols in clinical trials: Transparency and traceability of consent," *F1000Research*, vol. 6, no. 66, 2017. (doi:10.12688/f1000research.10531.5)

[2] C. H. Lee and K. Kim, "Implementation of IoT system using block chain with authentication and data protection," in *International Conference on Information Networking*, pp. 936-940, 2018.

[3] M. Fukumitsu, S. Hasegawa, J. Iwazaki, M. Sakai and D. Takahashi, "A Proposal of a secure P2P-type storage scheme by using the secret sharing and the blockchain," in *IEEE 31st International Conference on Advanced Information Networking & Applications*, pp. 803-810, 2017.

[4] X. Huang, C. Xu, P. Wang, *et al.*, "LNSC: A security model for electric vehicle and charging pile management based on blockchain ecosystem," *IEEE Access*, vol. 6, pp. 13565–13574, 2018.

[5] E. O. Kiktenko, N. O. Pozhar, M. N. Anufriev, *et al.*, "Quantum-secured blockchain," *Quantum Science and Technology*, vol. 3, no. 3, pp. 035004, 2018.

[6] H. W. Kim, Y. S. Jeong, "Secure Authentication-Management human-centric Scheme for trusting personal resource information on mobile cloud computing with blockchain," *Human-centric Computing and Information Sciences*, vol. 8, no. 1, pp. 1-13, 2018.

[7] J. H. Lee, "BIDaaS: Blockchain based ID as a service," *IEEE Access*, vol. 6, pp. 2274–2278, 2018.

[8] C. Lin, D. He, X. Huang, *et al.*, "BSeIn: A blockchain-based secure mutual authentication with fine-grained access control system for industry 4.0," *Journal of Network and Computer Applications*, vol. 116, pp. 42–52, 2018.

[9] Q. Lin, H. Yan, Z. Huang, *et al.*, "An ID-based linearly homomorphic signature scheme and its application in blockchain," *IEEE Access*, vol. 6, pp. 20632–20640, 2018.

[10] L. N. Lundbaek, A. C. D'Iddio, M. Huth, "Optimizing governed blockchains for financial process authentications," *Cryptography and Security*, 2016. arXiv:1612.00407.

[11] T. Salman, M. Zolanvari, A. Erbad, R. Jain and M. Samaka, "Security services using blockchains: A state of the art survey," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 1, pp. 858-880, 2018.

[12] H. Wang, Y. Song, "Secure cloud-based EHR system using attribute-based cryptosystem and blockchain," *Journal of Medical Systems*, vol. 42, no. 8, pp. 152, 2018.

[13] W. Yin, Q. Wen, W. Li, *et al.*, "An anti-quantum transaction authentication approach in blockchain," *IEEE Access*, vol. 6, pp. 5393–5401, 2018.

[14] R. Yu, J. Wang, T. Xu, *et al.*, "Authentication with block-chain algorithm and text encryption protocol in calculation of social network," *IEEE Access*, vol. 5, pp. 24944–24951, 2017.

[15] G. Zhitao, S. Guanlin, Z. Xiaosong, *et al.*, "Privacy-preserving and efficient aggregation based on blockchain for power grid communications in smart communities," *IEEE Communications Magazine*, vol. 56, no. 7, pp. 82–88, 2018.

Biography

Pengfei Fan, born in 1988-9-28, male, from Baotou, Inner Mongolia, China, has gained the master's degree. He is now working in State Grid Inner Mongolia Eastern Electric Power Co., Ltd. He is Network Security Engineer. He is interested network security.

Yazhen Liu, born in 1991-9-5, female, from Wuhai, Inner Mongolia, China, has gained the master's degree. She is now working in State Grid Inner Mongolia Eastern Electric Power Co., Ltd. She is Network Engineer. She is interested information communication network.

Jiyang Zhu, born in 1981-12-9, male, from Huludao, Liaoning Province, China, has reeceived an undergraduate degree. He is now working in State Grid Inner Mongolia Eastern Electric Power Co., Ltd. Information and Communication Branch. He is deputy director of Information Communication Operation and Maintenance Center. He is interested information communication networkand integrated management.

Xiongfei Fan, born in 1984-4-29, male, from Baotou, Inner Mongolia, China, has gained the master's degree. He is now working in training center of Inner Mongolia Electric Power (Group) Co., Ltd. He is a trainer by profession. He is interested dispatching automation.

Liping Wen, born in 1991-09-05, female, from Hohhot, Inner Mongolia, China, has gained the master's degree. She is now working in Inner Mongolia Electric Power (Group) Co., Ltd. Hohhot Power Supply Bureau. Her occupation is to monitor and inquire about electricity. She is interested Business Development.

Secure High Capacity Data Hiding Scheme based on Reference Matrix

Xiao-Shuang Li[1], Chin-Chen Chang[2], Ming-Xing He[1], and Chia-Chen Lin[3]
(Corresponding author: Chia-Chen Lin)

School of Computer and Software Engineering, Xi Hua University[1]
Department of Information Engineering and Computer Science, Feng Chia University[2]
Department of Computer Science and Information Management, Providence University[3]
Taichung 433, Taiwan
(Email: mhlin3@pu.edu.tw)

(Received July 17, 2018; Revised and Accepted Mar. 14, 2019; First Online Oct. 12, 2019)

Abstract

In this paper, a secure high-capacity data hiding scheme based on a reference matrix is proposed. With the help of the numbering reference matrix and a look-up table, each pixel pair of a cover image can conceal 6 secret bits, which offers 2 extra secret bits than Liu *et al.'s* scheme, while maintaining the average PSNR up to 41.97 dB. Experimental results confirm that our proposed scheme outperforms previous data hiding schemes in visual quality and hiding capacity. Moreover, statistical analysis confirms that the hidden data can be securely protected.

Keywords: Data Hiding; Hiding Capacity; Look Up Table; Reference Matrix

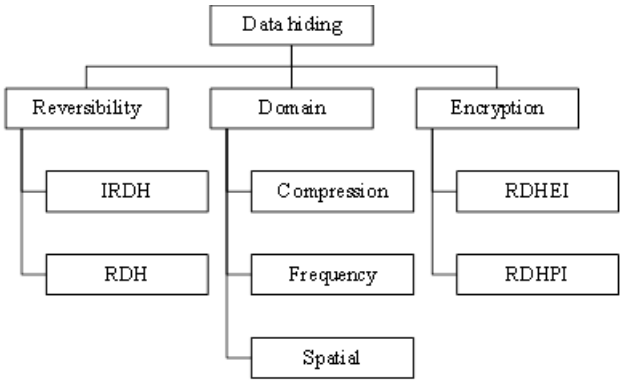


Figure 1: Taxonomy of data hiding

1 Introduction

With the development of information technologies, data hiding has attracted considerable researchers' attention in the field of information security because it can guarantee the security of the transmitted data over the Internet besides adopting the traditional cryptographic approaches, such as RSA [1], El Gamal [2], and DES [3]. The purpose of data hiding is to invisibly embed secret data into a cover medium, which can be audio, images, text, em etc. Similar to camouflage used by animals and insects to blend into the natural environment to protect themselves, the recognition of data that is hidden in cover images can be minimized when data hiding is adopted. To give a clear classification of the existing data hiding schemes, a taxonomy of data hiding is presented in Figure 1.

The first data hiding scheme was proposed by Bender *et al.* [1] in 1996. Over the next twenty years, many data hiding (DH) schemes have been proposed. DH schemes can be classified into three categories according to different criterion. For example, based on reversibility, DH schemes can be classified into reversible data hiding (RDH) [5, 9, 15, 16, 18, 25, 30] and irreversible data hiding (IRDH) [2, 3, 7, 11, 13, 17, 20, 21, 24, 27, 29]. The former reversible schemes are especially designed for military and medical applications and the original cover images can be restored after the hidden secret data is extracted. The latter irreversible schemes are considered conventional DH schemes, and the cover images can not be completely restored even the hidden secret data has been extracted.

The latest category is to apply data hiding to the encryption files, as first proposed in 2011 by Zhang [31]. The main idea of DH in an encryption file is to hide secret data into the encryption files so that data hiding applications can be expanded to areas such as healthcare, which emphasizes the confidentiality of patient information, and cloud applications that need to protect customer data which is stored at cloud service providers' side. Based on the encryption criteria, DH schemes can be classified into reversible data hiding in encrypted images (RDHEI) [5, 18, 25, 30] and reversible data hiding in public images (RDHPI) as shown in Figure 1. The former is to embed secret data into an encryption file and the latter is to embed secret data into a public image, such as a

Hello Kitty image.

Apart from reversibility and encryption criterion, DH schemes can also be classified into three subcategories: compression domain [5, 8, 9, 15], frequency domain [4, 14] and spatial domain [2, 3, 7, 11, 13, 17, 20, 21, 24, 27, 29], according to the domain where secret data is embedded. For the compression domain, a DH scheme can hide secret data into compression codes generated by various compression algorithms, such as BTC [28], VQ [12], SMVQ [10], *etc.* For the frequency domain, the secret data is embedded into the DCT [5, 9, 15] or DWT coefficients [14]. Take Chang *et al.'s* scheme [5] for example, they embedded secret data into the two successive zero DCT coefficients of the medium-frequency components in each block of the cover image.

For the spatial domain, secret data is directly embedded into the pixel value by modifying the pixel value according to the pre-determined hiding strategies [2, 3, 7, 11, 13, 17, 20, 21, 24, 27, 29]. The most famous DH scheme for the spatial domain are LSB-based DH schemes. Among them, the first simple LSB DH scheme was proposed by Chan and Cheng in 2004 [2], which used secret bits to replace the least significant bits of pixels in a cover image. Following Chan and Cheng's idea, many LSB-based DH variants have been proposed. For example, Mielikainen defined a binary function of two cover pixels, which is assigned to a pre-determined value [21]. With their design, a cover pixel pair becomes a unit during data embedding. The LSB of the first pixel carries one secret bit, and a function of the two pixel values also carries another secret bit.

To reduce the distortion caused by data embedding, besides LSB approach various hiding strategies have been proposed. Take Wu and Tsai's scheme for example [27], they used pixel value difference (PVD) to design their embedding strategy. Later, various PVD-based DH schemes were designed [11, 20, 24, 29] to increase hiding capacity while reducing distortion. The latest PVD-based DH scheme was proposed by Mehdi *et al.* [20] in 2017, in which parity-bit PVD is adopted to offer a high payload and good visual quality.

No matter what kind of embedding strategy is designed, there are usually complex computations involved when a DH scheme offers a higher hiding capacity, secure protection of the hidden data, and reduced distortion of cover image. In 2008, Chang *et al.* tried to propose a DH utilizing a Sudoku matrix to embed secret data to meet the above three requirements while reducing the computation cost [3]. Unfortunately, the visual quality of stego-image provided by Chang *et al.'s* scheme was not high. In order to improve the weakness of Chang *et al.'* scheme, Hong *et al.* [13] proposed a novel DH scheme by using a search algorithm. Later, in 2014, another new secret data hiding approach based on a turtle-shell reference matrix was proposed by Chang *et al.* [7] to offer good visual quality and enhance the hiding capacity without a high computation cost. Subsequently, based on the reference matrix and turtle shell concept, Liu *et al.* [17] defined a look-up table to allow a pixel pair to carry one extra bit than Chang *et al.'s* scheme [7].

Inspired by the schemes of Chang *et al.* [7] and Liu *et al.* [17], we aim to enhance both hiding capacity and visual quality while securely protecting the hidden data without a high computation cost. Later, experimental results will prove that in our method each pixel pair can conceal extra 2 secret bits compared to Liu *et al.'s* scheme and the visual quality of our proposed scheme is better than other previous schemes.

The rest of this paper is organized as follows. Section 2 briefly reviews related work. Section 3 explains our proposed scheme. Section 4 provides performance results and gives some discussions. Finally, a brief conclusion is given in Section 5.

2 Related Work

We review Chang *et al.'s* scheme [7] and Liu *et al.'s* scheme [17] in Subsections 2.1 and 2.2, respectively, to provide insight into how these works specifically inspired our scheme.

2.1 Chang *et al.'s* Turtle Shell Based DH Scheme

In 2014, a novel turtle-shell-based data hiding scheme was proposed by Chang *et al.* [7]. In Chang *et al.'s* scheme, a reference matrix M sized 256×256 digits, as shown in Figure 2, needs to be constructed first before the secret data is embedded into a cover image. Both the X and Y axes of reference matrix M represent the grayscale pixel values of an image and they are ranged from 0 to 255. Reference matrix M is composed of a large number of turtle shells. Each turtle shell is a hexagon shape and includes 6 edge elements and 2 back elements. Therefore, there are 8 different digits ranging from 0 to 7 in a turtle shell. In other words, three secret bits can be carried with a digit of the turtle shell.

Figure 2 illustrates an example of embedding secret data based on a reference matrix M. The location of each cover pixel pair (p_m, p_n) is mapped to (p_m, p_n) in reference matrix M and denoted as $M(p_m, p_n)$, where the p_m is the column value and the p_n is the row value. Assume the cover pixel pair is (4, 6) and the secret data is 7. $M(4, 6)$ belongs to the back element of a turtle shell, and its corresponding digit is 3, which is not equal to secret data 7. Therefore, the cover pixel pair $M(4, 6)$ is changed to $M(3, 5)$ because its corresponding digit is 7. In other words, the stego pixel pair is (3,5) to carry secret data 7. If the cover pixel pair is (6, 4) and the secret data is 2, then $M(6, 4)$ belongs to the edge element and its corresponding digit is 0 which is not equal to secret data 2. Since $M(6, 4)$ is the intersection point of three turtle shells, elements of three neighboring turtle shells need to be explored to find a pixel pair whose corresponding digit is equal to secret data 2. Finally, (6,4) is changed to (6,5)

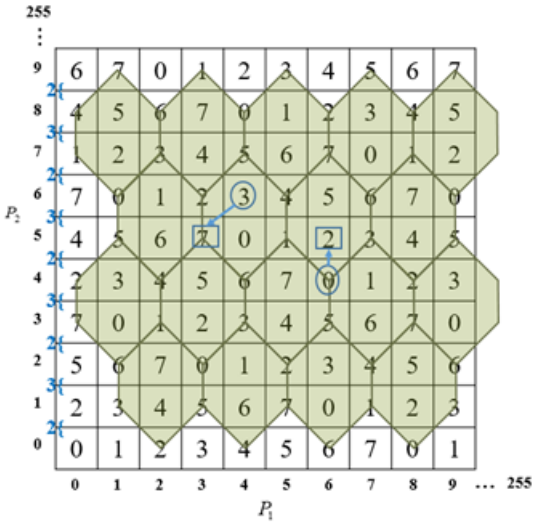


Figure 2: Examples of reference matrix M

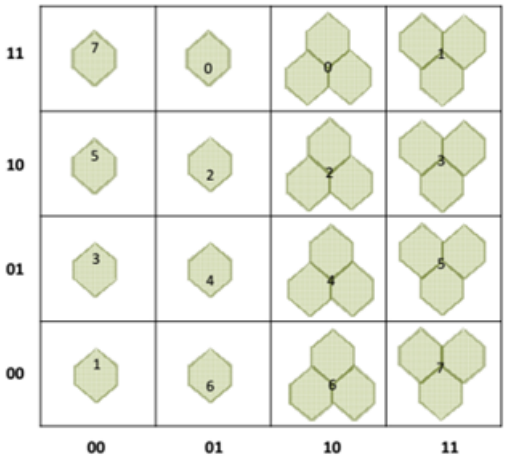


Figure 3: Location table T

to carry secret data 2 and the stego pixel pair is set as (6,5). Chang *et al.'s* idea is simple and computation cost of data embedding and data extraction is few.

2.2 Liu *et al.'s* High Capacity Turtle Shell Based DH Scheme

In 2015, Liu *et al.* also proposed a high capacity DH scheme based on turtle shells [17]. In their scheme, the reference matrix M is the same as that defined in Chang *et al.'s* scheme [7]. And reference matrix M and a location table T must be constructed in advance and the secret data stream is divided to non-overlapping 2 bits pieces. Each pixel pair of a cover image can embed 4 bits of secret data with the assistance of reference matrix M and location table T. Location table T shown in Figure 3 guides the modification policy of pixel pairs' values of the cover image during the data embedding phase and plays a crucial role to enhance hiding capacity.

Location table T defines 16 elements of the turtle shells as shown in Figure 3; however, they can be concluded as two different back elements of the turtle shell and two different edge elements. The definition of edge element is the same as that given in Chang *et al.'s* scheme [7], which is at the intersection of three neighboring turtle shells. Each element presented in reference matrix M is only mapped to a specific location defined in location table T. According to the architecture presented in reference matrix M, the values of the elements defined in location table T are always found in a set of values. For example, the values of the front back element defined in location table T are always in the set of values $\{1, 3, 5, 7\}$. Each element defined in location table T is represented by two indicators (p_j, p_j+1), where p_j and p_j+1 belong to $\{00, 01, 10, 11\}$, p_j is the column value, and p_j+1 is the row value in the location table T. Note that p_j and p_j+1 are two secret patterns.

During data embedding, 2-bit secret pieces are first mapped to location table T to find the specific pattern with a label. For example, (11, 01) is mapped to the edge element pattern with label 5. Once the pattern and its label are found, candidates with the same combination can be found from reference matrix M and be located. After that, the distance between elements which mapped to the original pixel pair and candidate can be calculated using Equation (1). The candidate with the minimal distance is then selected to carry the 2-bit secret pieces and its corresponding axes' values of reference matrix M is the pixel pair of the stego-image.

$$d(X, Y) = \sqrt{(X_i - Y_i)^2 + (X_j - Y_j)^2}, \tag{1}$$

where the (X_i, X_j) is the selected candidate, and the (Y_i, Y_j) is the original cover pixel pair. Afterwards, according to reference matrix M and location table T, the secret data can be successfully extracted.

Assume the original cover pixel pair is (4, 6), and the binary secret data is $(10\ 00)_2$. According to location table T, (10, 00) is mapped to the edge element with label 6. As Figure 4 shows, there are pairs M(4, 4), M(7, 6), and M(9, 2) with the same pattern and same label. Among them, only M(4,4) has the minimal distance with the pixel pair of the cover image; therefore, pixel pair (4,6) of the cover image is changed to (4,4) of the stego-image to carry secret bits (10 00).

2.3 Discussions

The schemes of both Chang *et al.* and Liu *et al.* used the same reference matrix M. In their reference matrix, each turtle shell only covers 8 elements ranging from 0 to 7. Chang *et al.* directly used 8 elements mapped to a turtle shell to carry secret data; therefore, the hiding capacity is limited to 3 secret bits. From Liu *et al.'s* scheme [17], we found they defined a location table T to first specify a pattern with a label, then candidates

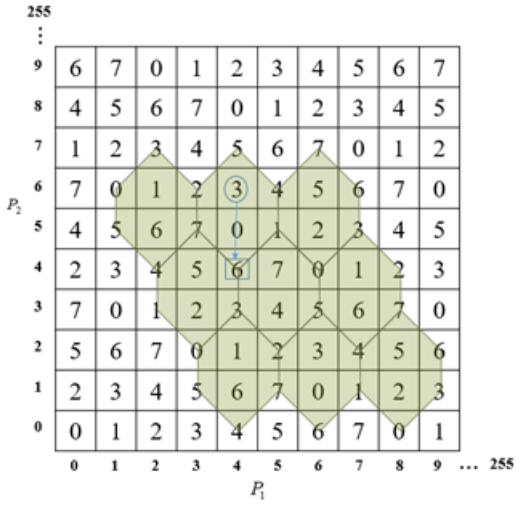


Figure 4: Example of data embedding with Liu *et al.'s* scheme

with the same pattern and label are found from reference matrix M. Only the candidate with the minimal distance from the pixel pair of the cover image can be selected as he pixel pair of the stego-image. With the assistance of location table T, the hiding capacity of a pixel pair is up to 4 bits, which offers one extra bit than Chang *et al.'s* scheme [7]. Thus, location table T can be treated as a new grouping and it allows a pixel pair of cover image to carry one extra secret bit compared with the scheme by Chang *et al.* .

3 The Proposed Secure High Capacity Scheme

Inspired by Chang *et al.* and Liu *et al.*, we found there are two ways to increase hiding capacity: one is to redefine a reference matrix; and the other is to define a new look-up table to offer the similar function as location table T did in Liu *et al.'s* scheme [17]. In our proposed scheme, we first define a turtle shell matrix (TSM), which is an upgraded version of reference matrix M as defined in Section 2.1, and a numbering reference matrix (NRM). Then, we added a look-up table which plays the role of location table T to enhance the hiding capacity. Definitions of TSM and NRM and related discussions are given in Subsections 3.1 and 3.2. Construction of the look-up table is described in Subsection 3.3. The embedding phase and extraction phase are given in Subsections 3.4 and 3.5, respectively.

3.1 Definitions of the Turtle Shell Reference Matrix (TSM) and the Numbering Reference Matrix (NRM)

Chang *et al.* scheme [7] defined a turtle shell as a hexagon shape with 8 different digits, which ranges from 0 to 7 as shown in Figure 2. According to their definitions, a turtle shell contains 2 back elements and 6 edge elements. To enhance the hiding capacity, we add 8 to number on Chang *et al.'s* reference matrix based on our pre-determined patterns, which are yellow circles indicated in Figure 5(b). Finally, a new turtle shell reference matrix (TSM) can be found, as shown in Figure 5(a). Comparing with Figures 5(a) and 5(b), it is noted that digits of that three neighboring turtle shells are ranged from 0 to 15 in the TSM rather than 0 to 8, this is because certain elements' values have been added with 8.

Once TSM is constructed, values of X axis and Y axis TSM are relabeling with 0 and 1 to derive a new reference matrix called numbering reference matrix (NRM) as shown in Figure 6. It is noted that NRM is based on TSM; therefore, both are the same size of 256×256.

3.2 NRM Numbering Rules

To further increase the hiding capacity of our proposed scheme, a new reference NRM must be generated and relabeled with two digits 0 and 1 as mentioned at the end of Subsection 3.1. However, there are many ways to label the values of the X axis and Y axis, such as, numbering the X axis in the order of (010101...) and the Y axis in the order of (11001100...) and so on. No matter what kind of numbering order is given, there is a crucial rule must be hold. That is, the numbering results must make sure the search scope is minimized. This is because different number order will lead the different size of search area. If the search area is larger, the distortion between the original pixel pair and the stego pixel pair will be larger, and it will lead to larger distortion of the stego-image. To maintain good visual quality, the numbering order which offers the minimal search area must be found. We experimented with various numbering strategies to find one offering a minimal search scope. For example, numbering the values of X axis in the order of (101010...) and numbering the values of Y axis in the order of (101010...), give the following numbering results as shown in Figure 7. We find with such a numbering strategy, taking digit 12 for example, only (01) and (11) mapped to digit 12 in the NRM. However, to form a new reference matrix, each digit presented in the NRM must map to four patterns (00), (01), (10), (11) so that each digit can carry 2 secret bits. If a numbering strategy such as numbering the values of X axis in the order of (101010...) and numbering the values of Y axis in the order of (101010...) is used, we can find that although the search scope has been expanded, digit 12 which maps to (00) and (10) patterns are still missing. As such, this numbering strategy cannot be used for the NRM.

After conducting many experiments, we found a numbering strategy which numbering the values of X axis in the order of (001100...) and numbering the values of Y axis in the order of (001100...) as shown in Figure 8. It is the best numbering result among all numbering strategies because it covers 4 combinations of 2 bits, and the

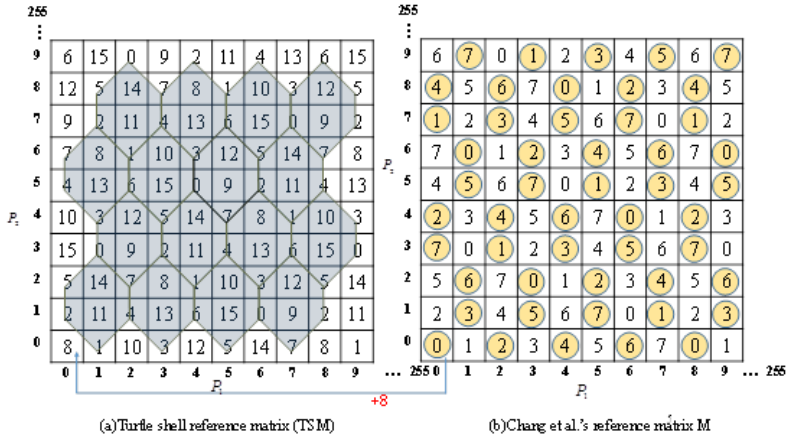


Figure 5: Turtle shell reference matrix (TSM)

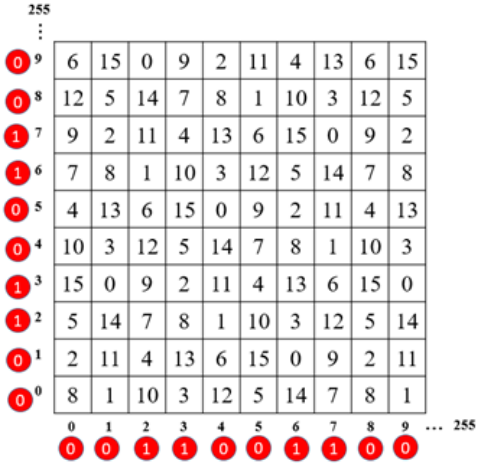


Figure 6: Numbering Reference Matrix (NRM)

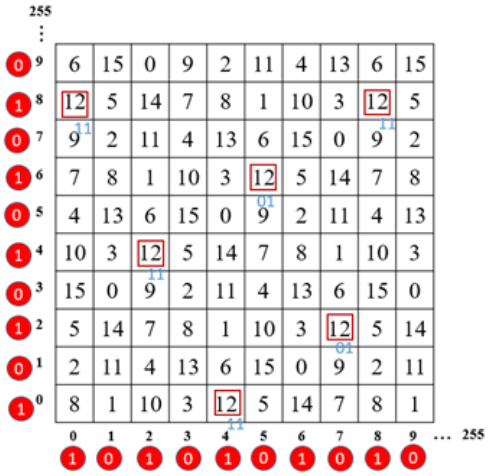


Figure 7: Example of NRM numbering strategy (101010...)

search area is always within the graphic area as shown in Figure 8.

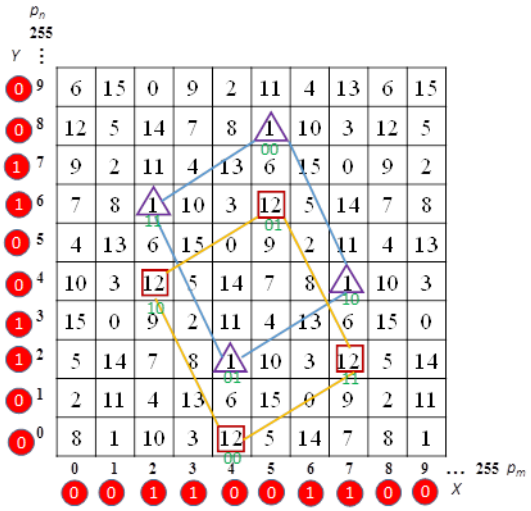


Figure 8: Optimal NRM numbering strategy (001100...)

3.3 Look-up Table Construction

To increase the embedding capacity, we designed a look-up table to carry 6 bits of secret data ranging from (000000) to (111111). As Figure 9 shows, each column contains 16 different digits of LU-values ranging from 0 to 15 and is listed from the bottom to the top. Each digit LU-value is transformed into a binary representation and presents in 4 bits and become the values of the Y axis of the look-up table as shown in Figure 9. For example, the LU-value located at (1,1) in the look-up table is equal to 15 and its binary stream is $(1111)_2$. The values of the X axis and Y axis of the numbering results from the NRM are combined as 2 bits and becomes the values of the X axis of the look-up table as shown in Figure 9.

Since the values of the Y axis of the look-up table are represented as 4 bits and the values of the X axis of look-up table are represented as 2 bits, there are 6 secret bits in total that can be represented by using our defined look-up table as shown in Figure 9. Certainly, multiple candidates can be found by combining our defined look-up table and NRM. To find a candidate which causes the least distortion between the original pixel pair and stego pixel pair, Equation (2) is defined to calculate the distance between the original cover pixel pair (A_m, A_n). and the candidate (B_m, B_n).

$$d(A,B) = \sqrt{(A_m - B_m)^2 + (A_n - B_n)^2}, \quad (2)$$

where (A_m, A_n) is the cover pixel pair mapping to NRM and (B_m, B_n) is the stego pixel pair mapping to NRM.

3.4 Embedding Phase

As we mentioned in the above subsections, the TSM, NRM and look-up table must be constructed before data embedding. For a grayscale cover image sized $H \times W$ pixels, which is composed by $H \times W$ pixels $P = \{p_m | m = 1, 2, \cdots, (H \times M)\}$, the secret message is divided into non-overlapping 6 bits, where the first 2 bits are mapped to the numbering results of values of the X axis and Y axis of NRM. The last 4 bits are mapped to the Y axis of the look-up table. In order to embed 6 bits secret data into a cover pixel pair (p_m, p_n), cover pixel pair (p_m, p_n) is mapped into the NRM first and denoted as NRM (p_m, p_n), where the p_m is the column decimal value and the p_n is the row decimal value and both p_m and p_n are ranged from 0 to 255. Next, an LU-value of the look-up table can be determined according to the last 4 bits of the Y axis of the look-up table. Find NRM (p_m, p_n) whose corresponding digit is equal to a pre-determined LU-value, where its numbering results are equal to the value of the X axis of the look-up table and the distance between NRM (p'_m, p'_n) and cover pixel pair NRM (p_m, p_n) is minimal. Finally, the a stego pixel pair is determined as (p'_m, p'_n).

In order to explain our proposed embedding phase more clearly, two examples regarding data embedding are demonstrated in Figure 10.

Example 1. *Assume cover pixel pair is (4, 6), the secret data is 9 and its binary representation is (00 1001)2=9. Using the last 4 bits as the indicator, we find the column which maps to (1001) and LU-value, which is 9 in the look-up table. Then, we find digit 9 and its corresponding renumbering results are (0,0) from the NRM. Once multiple candidates are found, the minimal distance between candidate and cover pixel pair (4,6) is applied to determine a candidate which causes less distortion. Here, NRM (5,5) is determined because it is the closest to cover pixel pair (4,6). Finally, NRM (4, 6) is modified to NRM (5,5). In other words, the cover pixel pair (4,6) is changed to stego pixel pair (5,5).*

Example 2. *Assume cover pixel pair is (3, 3), the secret data is 6 and its binary representation is (00 0110)2=6. We use (0110) as the indicator to find the column of the look-up table, which is 6 in the look-up table. From Figure 10, we find NRM (0,9), and NRM (4,1) whose digits are equal to secret data 6, and their numbering results are the same as (0,0). Therefore, the distance between NRM (0,9) and NRT (3,3), and distance between NRM (4,1) and NRM (3,3) is computed, respectively. Finally, NMR (4,1) is selected because it is closer to NRM (3,3) compared with NRM (0,9). The cover pixel pair (3,3) is changed to stego pixel pair (4,1).*

3.5 Extracting Phase

After embedding all of the secret message, the stego-image is generated. Once a receiver obtains the stego-image, the hidden secret data can be extracted with the assistance of NRM. To extract the hidden data, the receiver maps pixel pair of the stego-image into NRM of the NRM. Then, the receiver transforms the mapped digit into a binary representation and these bits are the last 4 bits of the hidden secret bits. According to the numbering results to which NRM maps, the receiver can get the first 2 secret bits. Finally, a secret unit can be derived by combining the above 2 bits and 4 extracted secret bits. The same operations are conducted continuously until all stego pixel pairs are processed and then the secret message can be extracted.

Example 3. *Take stego pixel pair (5, 5) as an example. The stego pixel pair maps to NRM (5,5) so that digit 9 and the numbering result (0,0) can be found. By transforming 9 into binary representation as (1001) and it means that the last 4 secret bits are (1001). As we mentioned, the numbering result (0,0) are the first 2 secret bits. Finally, the secret unit is derived as (00 1001)2 by combining above secret bits. Finally, a secret data 9 can be obtained after transforming (00 1001)2 into the decimal value. Take stego pixel pair (2, 7) for the other example. Stego pixel pair (2,7) maps to NRM (2,7) so that the digit 11 and the numbering result (1, 1) can be found from NRM. By transforming 11 into binary representation as (1011)2 and then combining (11) and (1011) as a new secret unit (11 1011)2. Transforming (11 1011)2 into the decimal value, receiver finally derives secret data as 59.*

4 Experimental Results

To prove the performance of the proposed scheme, several experiments are conducted. All experiments were implemented in Matlab 2012 on a PC with Intel(R) Core(TM) i7-3770 CPU 3.40 GHz, 8 GB RAM. Figure 11 shows the ten standard grayscale test images sized 512×512 that were used in our experiments: Wine, Lena, Harbour, Office, Airplane, Peppers, Baboon, Goldhill, Elaine, and Sailboat.

To estimate the visual quality of the stego-images we used the peak-signal-to-ratio (PSNR) as the measure-

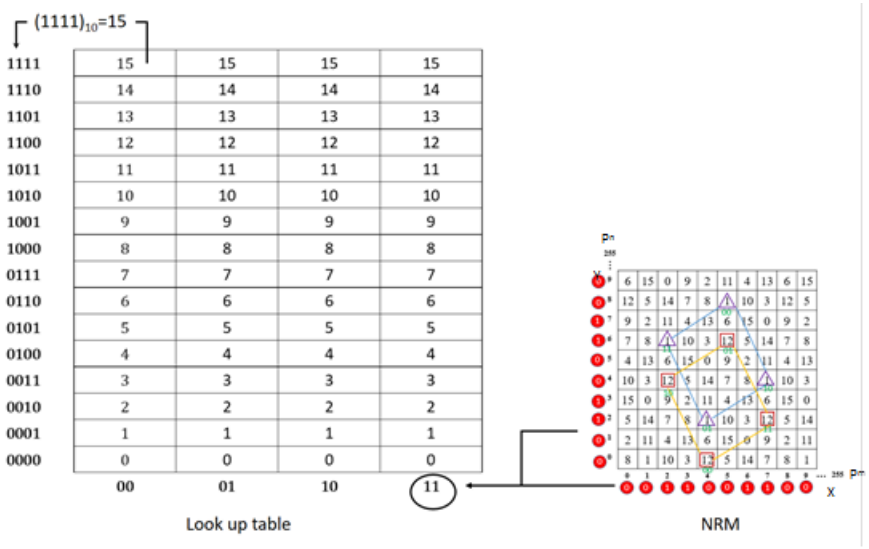


Figure 9: Look-up table

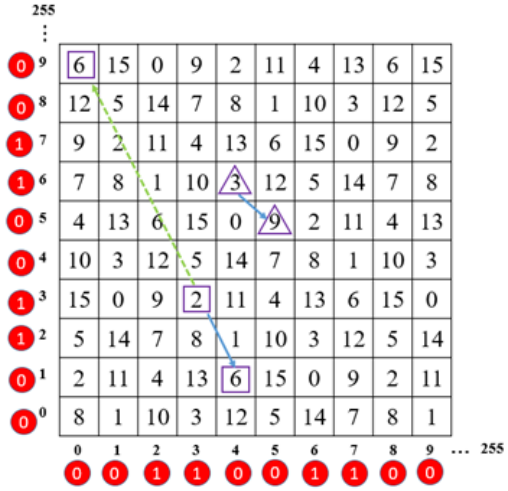


Figure 10: Example of our proposed embedding phase

ment. The definition for PSNR is given in Equation (3).

$$PSNR = 10\log_{10}(\frac{255^2}{MSE})(db), \tag{3}$$

where the mean square error (MSE) is between the original cover image and the stego-image, where for a grayscale cover image $H \times W$ pixels is defined as Equation (4):

$$MSE = \frac{1}{H \times M}\sum_{i=1}^{H\times M}(p_i - p_j)^2, \tag{4}$$

where p_i is the pixel value of the original cover image, and p_j is the pixel value of the corresponding stego-image. The higher the PSNR, the better the image quality. In general, if the PSNR is higher than 30 dB, it is difficult for the human eye to recognize any difference between the original cover image and stego-image. The stego-images generated with our proposed scheme carrying 786,432 secret bits are shown in Figure 12. The secret bits used in our experiments are randomly generated bitstream.

In addition to the PSNR, the Structural Similarity Index Metric (SSIM) is measured the degradation in the quality, which is based on structural information. And the value of SSIM is between -1 to 1. The value of 1 means the two images are identically the same. SSIM between the original image I and the corresponding stego-image C is defined as Equation (5):

$$SSIM(I,C) = \frac{(2u_Iu_C + c_1)(2\sigma_{IC} + c_2)}{(u_I^2 + u_C^2 + c_1)(\sigma_I^2 + \sigma_C^2 + c_2)}, \tag{5}$$

where u_I, u_C, σ_I^2, σ_C^2 are the averages and variances of I and C respectively, σ_{IC} is the covariance between I and C, c_1 and c_2 are as follows:

$$c_1 = (k_1L)^2; \quad \text{where } k_1 \ll 1 \text{ (small constant)}, \tag{6}$$
$$c_2 = (k_2L)^2; \quad \text{where } k_2 \ll 1 \text{ (small constant)}, \tag{7}$$

where L is defined as the dynamic range of the pixel values.

In addition, we also calculate the Normal Cross Correlation (NCC) between the original image I and the corresponding stego-image C as defined in as Equation (8):

$$NCC = \frac{\sum_i\sum_j I_{ij}C_{ij}}{\sum_i\sum_j (I_{ij})^2} \tag{8}$$

where the I_{ij} and C_{ij} are the original and stego-image bits at $(i,j)^{th}$ position. When the value of NCC is 1, it indicates two images are identically the same, and vice versa.

Figure 11 shows ten original cover images $a, b, c, d, e, f, g, h, i, j$, and Figure 12 shows the corresponding stego-images $a', b', c', d', e', f', g', h', i', j'$. Even though the

Figure 11: Ten 512×512 graysacle test images

Figure 12: The stego-images corresponding the ten test grascale images

amount of the secret message is up to 786,432 bits, the average visual quality of the stego-images is higher than 41 dB, and the average of the SSIM is 0.9342, and the least of the NCC is 0.9991 (See Table 1).

To better demonstrate the advantages of our proposed scheme, we also compared our scheme with previous schemes, such as those by Yang *et al.* [29], Liu *et al.* [17] and Shen and Huang [24] and Mehdi *et al.* [20]. The comparison results are listed in Table 2. Obviously, the maximum embedding capacity of our proposed scheme is much higher than the other schemes, and moreover the visual quality is higher than that of Yang *et al.'s* [29], Shen and Huang's [24] and Mehdi *et al.'s* [20]. Especially, the average embedding capacity of our proposed scheme surpasses the 356,209 bits of the Yang *et al.'s* scheme, exceeds the embedding capacity of the scheme of Shen and Huang by 372,943 bits, and also exceeds the 551,491 bits of the Mehdi *et al.'s* scheme. Although Liu *et al.'s* average PSNR is higher by 3.59 dB than our scheme, the hiding capacity is still 524,288 bits, which is 262,144 fewer secret bits compared to our scheme.

To demonstrate the visual quality performance of our proposed scheme, the third experiment was conducted and the comparisons among our proposed scheme and three previous schemes which claimed they can offer better image quality of setgo image or provide high hiding capacity [11, 17, 20] are shown in Table 3. Here, all schemes carried the similar amount of secret data to derive the PSNR values. Table 3 shows that the visual quality of our proposed scheme is better than the other three previous schemes. Note that the average PSNR of Liu *et al.'s* scheme is 45.55 dB, which is lower than that of our scheme 46.82 dB when the hiding capacity is set as 524,288 bits. By combining Tables 2 and 3, it is confirmed that our proposed scheme has a higher hiding capacity and offers better visual quality in the stego-image with the same hiding capacity.

To further prove the safety of our proposed scheme, we examined the pixel value difference (PVD) histograms of the original cover images and corresponding stego-images, where both are at their maximum embedding capacity as shown in Figure 13. The PVD histogram is calculated by computing the difference in the neighboring pixels between the original cover image and the stego-image. The smaller the gap between the two curves, the smaller the image changes, which confirms that the stego-image is more secure. Using the test images 'Baboon' and 'Peppers' for example, we show their PVD histograms after completely embedding secret data. As Figure 13 shows, the gap between two curves is small for the two test images. This confirms that our proposed scheme offers a relatively high visual quality and also guarantees the security of the hidden data.

To prove the computation cost is still low even the distance between the original cover pixel pair and multiple candidates pixel pairs need to be computed to reduce the potential distortion caused during data embedding. The computation time of data embedding and data extracting phases are listed in Table 4. From Table 4, we can see the proposed scheme is quite efficient and suitable for real-time applications.

5 Conclusions

In this paper, a novel data hiding scheme based on reference matrix and look-up table is proposed. The use of a NRM and look-up table not only allows 6 secret bits to be conceled in a pixel pair of the cover image, but also successfully reduces the caused distortion during data embedding. During extracting phase, only NRM is required; therfore, the extraction phase is also quite efficient. Lastly, the experimental results confirmed that our proposed scheme offers higher embedding capacity than other existing schemes while maintaining good visual quality and guaranteeing the security of the hidden data.

Acknowledgments

This research is funded by: (1) National Natural Science Foundation of China (nos.U143310218); (2) Chunhui Project of Education Ministry of China (nos.Z2014045); (3) Science and Education and Technology Bureau Project of Cheng-du Municipality (nos.2016-XT00-00015-GX); (4) Graduate Innovation Fund Project of Xi Hua University (nos.ycjj2018003).

References

[1] W. Bender, D. Gruhl, N. Morimoto, "Techniques for data hiding," *IBM Systems Journal*, vol. 35, no. 3, pp. 313–336, 1996.

[2] C. K. Chan, L. M. Cheng, "Hiding data in images by simple LSB substitution," *Pattern Recognition*, vol. 37, pp. 469–474, 2004.

[3] C. C. Chang , Y. C. Cho, T. D. Kieu, "An information hiding scheme using sudoku," in *Proceedings of Third International Conference of Innovative Computing, Information and Control*, pp. 17–22, 2008.

[4] C. C. Chang, T. D. Kieu, Y. C. Chou, "A lossless data embedding technique by joint neighboring coding," *Pattern Recognition*, vol. 42, no. 7, pp. 1597–1603, 2009.

[5] C. C. Chang, C. C. Lin, C. S. Tseng, W. L. Tai. "Reversible hiding in DCT-based compressed images," *Information Sciences*, vol. 177, no. 13, pp. 2768–2786, 2007.

[6] C. C. Chang, P. Y. Lin, Z. H. Wang, and M. C. Li, "A sudoku-based secret image sharing scheme with reversibility," *Journal of Communications*, vol. 5, no. 1, pp. 5–12, 2010.

[7] C. C. Chang, Y. J. Liu, T. S. Nguyen, "A novel turtle shell based scheme for data hiding," in *Proceedings*

Table 1: Experimental results of the proposed scheme

Images	EC (bits)	PSNR (dB)	BER	SSIM	NCC
Wine	786,432	41.96	0	0.9221	0.9989
Lena	786,432	41.96	0	0.9210	0.9991
Harbour	786,432	41.94	0	0.9258	0.9993
Office	786,432	41.97	0	0.9125	0.9993
Airplane	786,432	41.97	0	0.9206	0.9994
Peppers	786,432	41.97	0	0.9334	0.9991
Baboon	786,432	41.95	0	0.9657	0.9991
Goldhill	786,432	41.96	0	0.9531	0.9991
Elaine	786,432	41.96	0	0.9550	0.9992
Sailboat	786,432	41.97	0	0.9327	0.9992
Average	786,432	41.96	0	0.9342	0.9992

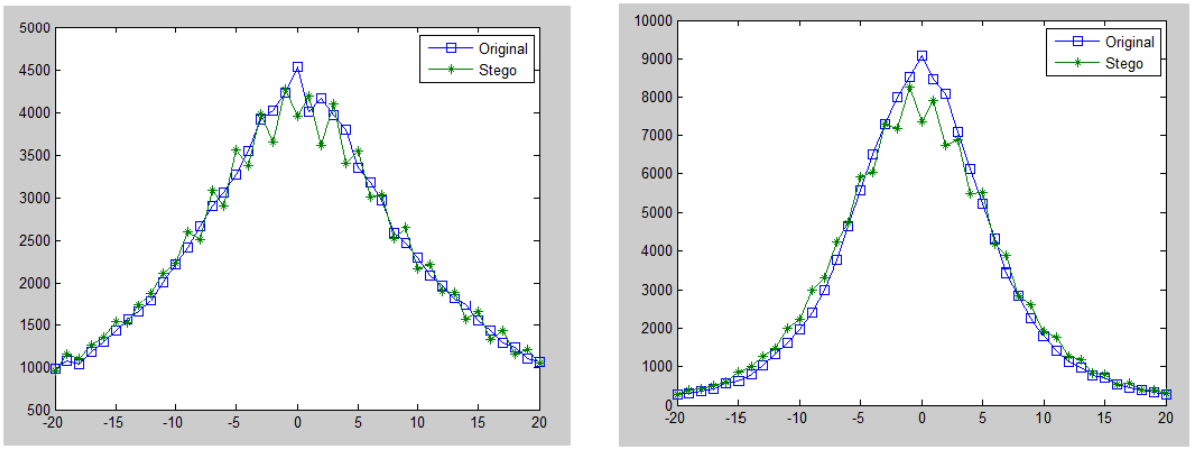
Note: EC= hiding capacity

Table 2: Comparisons the maximum EC and PSNR of proposed scheme with four existing schemes

	[29]		**[20]**		**[17]**		**[24]**		**Proposed scheme**	
Images	EC	PSNR	EC	PSNR	EC	PSNR	EC	PSNR	EC	PSNR
Baboon	482,515	34.67	540,850	40.66	524,288	45.55	443,472	38.88	786,432	41.95
Peppers	408,281	40.47	587,971	41.14	524,288	45.54	404,226	41.25	786,432	41.97
Goldhill	418,575	40.25	536,210	41.10	524,288	45.58	405,956	41.81	786,432	41.96
Sailboat	430,888	38.11	539,652	41.45	524,288	45.54	411,306	41.29	786,432	41.97
Lena	410,854	40.54	552,773	39.44	524,288	45.55	402,485	42.46	786,432	41.96
Average	430,223	38.81	551,491	40.76	524,288	45.55	413,489	40.94	786,432	41.96

Table 3: PSNR comparison of proposed scheme with three schemes

	[17]		**[20]**		**[11]**		**Proposed scheme**	
Images	EC	PSNR	EC	PSNR	EC	PSNR	EC	PSNR
Lena	524,288	45.55	540,850	40.66	512,384	43.01	524,288	46.83
Baboon	524,288	45.55	587,971	41.14	512,516	43.58	524,288	46.82
Peppers	524,288	45.54	536,210	41.10	512,392	43.62	524,288	46.82
Elaine	524,288	45.54	539,652	41.45	493,520	43.41	524,288	46.80
Sailboat	524,288	45.55	552,773	39.44	524,508	42.86	524,288	46.83
Average	524,288	45.55	551,491	40.76	511,064	43.29	524,288	46.82



a. Baboon b. Peppers

Figure 13: PVD histogram of the original cover image and corresponding stego-image

Table 4: Execution time (s) of data embedding and data extracting phase

Images	Embedding Time (s)	Extraction Time (s)
Wine	1.0213	1.0123
Lena	1.0345	1.0325
Harbour	1.0246	1.0432
Office	1.0436	1.0256
Airplane	1.0325	1.0364
Peppers	1.0245	1.0532
Baboon	1.0325	1.0267
Goldhill	1.0248	1.0365
Elaine	1.0356	1.0245
Sailboat	1.0267	1.0542
Average	1.0301	1.0345

of Tenth International Conference of Intelligent Information Hiding and Multimedia Signal Processing (IIHMSP'14), pp. 89–93, 2014.

[8] C. C. Chang, T. C. Lu, "A difference expansion oriented data hiding scheme for restoring the original host image," *Journal of Systems and Software*, vol. 79, no. 12, pp. 1754–1766, 2006.

[9] C. C. Chang, R. Tang, C. C. Lin, W. L. Lyu, "High-capacity reversible data Hiding method for JPEG images," *Journal of Software*, vol. 13, no. 1, pp. 1–17, 2018.

[10] C. C. Chen and C. C. Chang, "High capacity SMVQ-based hiding scheme using adaptive index," *Signal Processing*, vol. 90, no. 7, pp. 2141–2149, 2010.

[11] J. Chen, "A PVD-based data hiding scheme with histogram preserving using pixel pair matching," *Signal Processing Image Communication*, vol. 29, no. 3, pp. 375–384, 2014.

[12] H. M. Feng and J. H. Horng, "VQ-based fuzzy compression systems designs through bacterial foraging particle swarm optimization algorithm," in *Proceedings of the 5th International Conference on Genetic and Evolutionary Computing*, pp. 256–259, 2011.

[13] W. Hong, T. S. Chen, C. W. Shiu, "A minimal Euclidean distance searching technique for Sudoku steganography," in *Proceedings of International Symposium of Information Science and Engineering*, pp. 515–518, Dec. 2008.

[14] S. Lee, C. D. Yoo, T. Kalker, "Reversible image watermarking based on integer-to integer wavelet transform," *IEEE Transactions on Information Forensics and Security*, vol. 2, no. 3, pp. 321–330, 2007.

[15] Y. K. Lin, "A data hiding scheme based upon DCT coefficient modification," *Computer Standards & Interfaces*, vol. 36, no. 5, pp. 855–862, Sept. 2014.

[16] Y. J. Liu, C. C. Chang, "A turtle shell-based visual secret sharing scheme with reversibility and authentication," *Multimedia Tools and Applications*, vol. 77, no. 19, pp. 1–16, 2018.

[17] Y. J. Liu, C. C. Chang, T. S. Nguye, "High capacity turtle shell-based data hiding," *IET Image Processing*, vol. 10, no. 2, pp. 130–137, 2015.

[18] K. Ma, W. Zhang, X. Zhao, N. Yu, and F. Li, "Reversible data hiding in encrypted images by reserving room before encryption," *IEEE Transactions on Information Forensics Security*, vol. 8, no. 3, pp. 553–562, Mar. 2013.

[19] M. Matsui, "Linear cryptanalysis method for DES cipher," in *Advances in Cryptology (Eurocrypt'93)*, LNCS 765, pp. 386–397, Springer, 1993.

[20] H. Mehdi., W. A. W. Ainuddin, T. S. Anthony Ho, J. Noman, "A data hiding scheme using parity-bit pixel value differencing and improved rightmost digit replacement," *Signal Processing Image Communication*, vol. 50, pp. 44–57, 2017.

[21] J. Mielikainen, "LSB matching revisited," *IEEE Signal Processing Letters*, vol. 13, no. 5, pp. 285–287, May 2006.
[22] C. Qin, C. C. Chang, and T. J. Hsu, "Reversible data hiding scheme based on exploiting modification direction with two steganographic images," *Multimedia Tools and Applications*, vol. 74, no. 15, pp. 5861–5872, 2015.
[23] R. L. Rivest, A. Shamir and L. Adleman, "A method of obtaining digital signatures and public key cryptosystems," *Communications of the ACM*, vol. 21, no. 2, pp. 120–126, 1978.
[24] S. Y. Shen., L. H. Huang, "A data hiding scheme using pixel value differencing and improving exploiting modification directions," *Computer & Security*, vol. 48, pp. 131–141, 2015.
[25] A. Smita, K. Manoj, "Mean value based reversible data hiding in encrypted images," *Optik*, vol. 130, pp. 922–934, 2017.
[26] Y. Tsiounis, M. Yung, "On the security of El Gamal based encryption," in *International Workshop on Public Key Cryptography*, LNCS 1431, pp. 117–134, Springer, 1998.
[27] D. C. Wu, W. H. Tsai, "A steganographic method for images by pixel-value differencing," *Pattern Recognition Letters*, vol. 24, no. 9, pp. 1613–1626, 2003.
[28] Y. G. Wu, S. C. Tai, "An efficient BTC image compression technique," *IEEE Transactions on Consumer Electronics*, vol. 44, no. 2, pp. 317–325, 1998.
[29] H. Yang, C. Y. Weng, H. K. Tso, *et al.*, "A data hiding scheme using the variet ies of pixel-value differencing in multimedia images," *Journal of Systems and Software*, vol. 84, no. 4, pp. 669–678, 2011.
[30] X. Zhang, Z. Qian, G. Feng, and Y. Ren, "Efficient reversible data hiding in encrypted images," *Journal of Visual Communnications and Image Represent*, vol. 25, no. 2, pp. 322–328, Feb. 2014.
[31] X. P. Zhang, D. Schonberg, and K. Ramchandran, "Reversible data hiding in encrypted image," *IEEE Signal Processing Letters*, vol. 18, no. 14, pp. 255–258, Feb. 2011.

Xiao-Shuang Li received her BS and MS degree in applied Information and Computing Science in 2016 and 2019 respectively from Xihua University, Chengdu, Sichuan, China. Her current research interests include data hiding, modern cryptographic algorithms, cloud computing security technologies and visual cryptography.

Chin-Chen Chang received his BS degree in applied mathematics in 1977 and the MS degree in computer and decision sciences in 1979, both from the National Tsing Hua University, Hsinchu, Taiwan. He received his Ph.D in computer engineering in 1982 from the National Chiao Tung University, Hsinchu, Taiwan. Since February 2005, he has been a Chair Professor of Feng Chia University. In addition, he has served as a consultant to several research institutes and government departments. His current research interests include database design, computer cryptography, image compression and data structures.

Ming-Xing He received the M.Sc Degree from Chongqing University and the Ph.D from Southwest Jiaotong University in 1990, 2003 respectively. He is a full professor of the School of Computer and Software Engineering, Xihua University, Chengdu, P.R.China. His current research interests include cryptography and information security. He has co-authored five books and has published over 100 papers in refereed professional journals and international conferences. He received the DAAD scholarship reward of Germany in 2002, the Excellent Ph.D. Dissertation Award in Southwest Jiaotong University in 2003, and the grant of National Science Foundation of China (NSFC) in 2004, 2007 and 2015. He is a Senior Member of CACR and member of the ACM.

Chia-Chen Lin (also known as Min-Hui Lin) received her Ph.D degree in information management in 1998 from the National Chiao Tung University. Dr. Lin is currently a professor of the Department of Computer Science and Information Management, Providence University. Since 2018, she is the Fellow of IET. In additions, she serves Associate Editor and Editor for several representative EI, SCIE journals. Her research interests include image and signal processing, information hiding, mobile agent, and electronic commerce.

Subgroup Operations in Identity Based Encryption Using Weil Pairing for Decentralized Networks

N. Chaitanya Kumar, Abdul Basit, Priyadarshi Singh, and V. Ch. Venkaiah
(Corresponding author: Abdul Basit)

School of Computer and Information Sciences, University of Hyderabad
Hyderabad-500046, India
(Email: abdulmcajh@gmail.com)
(Received June 10, 2018; Revised and Accepted Aug. 18, 2018; First Online Jan. 14, 2019)

Abstract

One of the drawbacks of the conventional public key systems is that the sender must know the public key of the recipient in advance for the key setup and retrieval. This problem can be solved in Identity Based Encryption (IBE) by taking some identifier string (*e.g.* an e-mail or phone number, *etc.*) as the public key. When a user wants to send a message then he only has to know this identifier string. The receiver requests the private key from a Trusted Third Party called PKG (Private Key Generator) to decrypt the message. The job of the PKG can be decentralized using the Shamir secret sharing scheme. The Weil Pairing on the elliptic curve is suitable to implement IBE, as it is based on bilinear maps between groups. In this paper, we propose a scheme that allows threshold decryption involving a subgroup of participants of the network.

Keywords: Identity Based Encryption; Subgroup Operations; Weil Pairing

1 Introduction

Identity Based Encryption (IBE) will allow the sender to use the receiver's identity in order to encrypt the message instead of using his public key. The usage of identity instead of public key has wide range of applications. The identity based encryption system uses an arbitrary string as an identity. The identity based encryption system is first developed by Shamir in 1984 [20] to simplify the management of certificates in an e-mail system. For example, when A wants to send a mail to B at B123@company.com, A encrypts the message simply by using B123@company.com. With this there is no need for A to obtain public key certificate of B. When B receives the mail then B contacts Private Key Generator (PKG) a third party organization and obtains the private key by authenticating himself. Finally, B can read the mail which was sent by A. Weil pairing is a mapping of two computational Diffie-Hellman groups where one group being hard. Initially Weil pairing was used to attack elliptic curve systems [17]. Later, Joux [11] designed a protocol using one round diffie-hellman key exchange among three parties and proved that weil pairing can also be used for good. Sakai *et al.* [19] also used weil pairing for the exchange of keys. Operations performed among the sub group of users belonging to a network and how they deal when a new user wants to be part of the network is known as Subgroup operations. Our proposed scheme demonstrates a protocol for subgroup operations and also decentralizes the job of PKG. The advantage of PKG being decentralized is that the communication becomes secure, more reliable when compared to existing systems. It also allows the new users to have the same abilities as that of the initial users and each user has their share for the remaining life of the network.

2 Preliminaries

2.1 Shamir Secret Sharing

The secret sharing mechanism shares the secret s among a group of participants $P = \{p_1, p_2, \cdots, p_n\}$ of n parties by using a special figure called dealer. The dealer sends privately the share of a secret to each party. Reconstruction process is adopted by the authorized subsets to extract the secret s from the given shares. The group of such authorized subsets are called as access structure. Shamir secret sharing scheme [21] uses the Lagrange's interpolation polynomial to implement (t,n) access structure where t is the threshold value and n is the no.of participants. For example let us consider n participants, s is the secret, t is the threshold and the finite field is denoted by F_p. Shamir secret sharing scheme has two phases namely: Distribution and reconstruction [2]. In the construction phases shares are distributed to the users and in the reconstruction phase the users compute the secret from their shares.

2.2 Elliptic Curve Cryptography

In cryptography, elliptic curve is defined over a finite field that contains all the points satisfying equation $y^2 = x^3+ax+b$ where $4a^3+27b^2 \neq 0$ along with a distinguished point at infinity denoted by O. The ECC security depends on the difficulty of elliptic curve discrete logarithm problem [10]. The Elliptic Curve Cryptosystems are hard under the discrete logarithmic problem which play a vital role in its security.

2.3 Weil Pairing

The Weil pairing is used to construct admissible pairings that can be used as the basis for cryptographic systems. Let us consider p as a prime number and is given by $p = 12q - 1$ for some random prime q. Let $y^2 = x^3 + 1$ be a super singular elliptic curve(E) over a finite field F_p. A cyclic group having order as $p + 1$ is formed by a group of rational points given by: E(F_p)=$\{(x, y) \in F_pXF_p : (x, y) \in E\}$. Now, as p+1=12q. There is a cyclic subgroup G_1 of order q. Let us consider G as a generator for G_1 and G_2 be the subgroup containing the elements having order q.

2.4 Identity Based Encryption

The identity based encryption schemes were first proposed by Shamir in [22] which is not practical in its approach. Later, Boneh and Franklin [5] proposed a scheme on identity based encryption which was secure and practical. Their scheme efficiently used the concept of bilinear mapping among groups which plays a vital role in our work. The identity based encryption scheme consists of four algorithms. They are: Setup, Extract, Encrypt and Decrypt.

1) Setup: In this algorithm, the system parameters are made public where as the master-key is known to Private Key Generator (PKG). This phase initially takes a security parameter as an input and returns the system parameters and master-key.

2) Extract: This algorithm obtains private key from the given public key.This algorithm takes the input parameters, arbitrary ID $\in \{0, 1\}^*$ and master key as input and return d as output. Here ID is a random string that is used as a public key and d is the private key which will be used later for decryption.

3) Encrypt: This algorithm takes the input parameters, message, ID as input and returns the ciphertext.

4) Decrypt: This algorithm takes the input parameters, cipher-text and d (private key) as input and returns the correspond message.

One of the main concern of IBE is to distribute the role of an authority or a trusted third party among the users. As a result there were many schemes proposed which adopted the secret sharing techniques. Zhou and Haas [26] were the first to propose such a scheme using the concept of threshold cryptography which is not that practical in its approach. Later Kong *et al.* [16] proposed another scheme but it was insecure. Other works [12, 18] distribute only a part of master key in identity-based environments. All of the above works use shamir secret sharing scheme and whenever a new user wants to be part of the network it imposes certain limitations like having a lot of interaction with existing users or not having the same ability as compared to other users. Blundo *et al.* [4] proposed a scheme in which new users can join the network dynamically without the need of any authority by using bivariate polynomials. Some other works Anzai *et al.* [1] and Daza *et al.* [6] used bivariate polynomials to decentralize the role of trusted authority.

2.5 Decentralization

In identity based encryption the master key is stored at the PKG and should be protected. To achieve this we will be distributing the master key among several users by using the concept of threshold cryptography. The users exchange the bivariate polynomial to decentralize the work of PKG. When working in subgroups it is suggested to work in small subgroup of a curve in order to increase the performance of an IBE system. Here we use Weil pairing to decentralize the PKG. In this system, public key of each user is transformed to a point on the group by hashing the ID to a point which is on the curve and later the point is multiplied by a constant.

3 Proposed System

In our system the role of PKG is fully decentralized as discussed in Section 3.2. After the initial exchange of polynomials each user has a share of a secret. He can communicate with other users or can perform subgroup operations using the given protocol.

3.1 Setup

Let L denote the initial set of N users in the network. This initial N users are known as founding users of the network. All those users will run the protocol designed in the initialization phase (specified in Subsection 4.2). The main goal is to decentralize the role of the PKG by using Shamir's secret sharing scheme, weil pairing and identity based encryption. Groups G_1, G_2 are taken for pairing each of them having a hash function. Threshold values t and t^1 are used for performing subgroup operations.

3.2 Initialization

Our scheme will have the following parameters which are made public. A group G which is additive of a prime order q and produced by a random point P under the assumption that the discrete logarithm problem is hard. In addition to the above, a bilinear pairing and two hash

functions are made public, bilinear pairing $e : G \times G \rightarrow G_T$, hash function $h : \{0,1\}^* \rightarrow Z_q$, hash function $H : \{0,1\}^* \rightarrow G$). Two threshold values t, t^1 are chosen, where the threshold value t will be used in significance to test the security of the designed network *i.e.* it will test that maximum $t-1$ nodes are deceptive. Another threshold value t^1 is used for looking after the security of the threshold operations computed in the users subgroup.The required condition for security is $t^1 \leq t \leq L$.
The bilinear pairing e and hash function H are needed to generate the individual keys based on identity or when we want to compute the threshold operations on subgroup of users. Initialization phase of our designed algorithm is described below:

1) Each user in L choses a random bivariate polynomial $F_i(x,z) \in Z_q[x,z]$ with degree utmost $t-1$ in the variable x and z. Here, L denote the initial set of N users in the network. Each polynomial $F(x,z) = \sum_{L_i \in L} F_i(x,z)$ (Here, L_i is the i^{th} user in given initial set of N users) share the same properties. The constant term of the given polynomial is $f_{i,0} = F_i(0,0)$.

2) Each user $L_i \in L$ secretly sends the bi-variate polynomial to the other users $L_j \in L$(founding users) in the form of $F_{ij}(x) = F_i(x, h(L_j))$. Later, user L_i computes $Y_i = f_{i,0}P$ and uses this value in every message.

3) After each user in L performs the above step, each user L_j will compute their final secret value and is given by:

$$\begin{aligned} S_j(x) &= \sum_{L_i \in L} F_j(x) \\ &= \sum_{L_i \in L} F_i(x, h(L_j)) \\ &= F(x,z). \end{aligned}$$

Each user computes their public key and make it public based on the information received from the other users $L_j \in L$. The public key(PK) will be as follows:

$$\begin{aligned} PK &= sP \\ &= \sum_{L_i \in L} f_{i,0}P \\ &= \sum_{L_i \in L} Y_i. \end{aligned}$$

Note: Implicitly secret key(s) is F(0,0). A share $[s_j] = s_j(0) = F(0,0) = F(0, h(L_j))$ of the secret key can be computed by each user in L_j from its partial information $S_j(x)$.This set up runs securely only when $t \leq L$.

3.3 Network Management

After the initialization phase is completed. If a new user N_k desires to be part of the network then he should run the below steps:

1) The new user N_k will select a group L_m which consists minimum of t users in the network and request them to include him in their Network.

2) If any of the user in L_m (suppose N_j) agrees to include this new user(N_k) in their network then he sends the following value:

$$\begin{aligned} S_j(h(N_k)) &= F(h(N_K), h(N_j)) \\ &= F(h(N_j), h(N_k)) \\ &= S_k(h(N_j)). \end{aligned}$$

3) When the new user N_k gets this information from t users then he uses Lagrange interpolation to extract secret polynomial as follows:

$$\begin{aligned} & \sum_{N_k \in L_m} \prod_{N_i \in L_m, i \neq j} \frac{x - h(N_i)}{h(N_j) - h(N_i)} S_j(h(N_k)) \\ = & \sum_{N_k \in L_m} \prod_{N_i \in L_m, i \neq j} \frac{x - h(N_i)}{h(N_j) - h(N_i)} F(h(N_j), h(N_k)) \\ = & F(x, h(N_k)) \\ = & S_k(x). \end{aligned}$$

4) Finally the share $[s_k] = S_k(0)$ is computed by N_k.

3.4 Secure Communication Using IBE

In IBE, the public key is derived directly from the identity of nodes in L_m *i.e.* $pk_m = H(L_m) \in G$ where $H : \{0,1\}^* \rightarrow G$ which is chosen as hash function during initialization phase. Since it is a decentralized network, the user N_k needs to contact other users to compute the secret key sk_m=sH(L_m) where the master secret key is s. The designed protocol is as follows:

1) The user N_k approaches a group of users (L_m) having minimum of t users to request for their share.

2) If any of the user (N_j) in the group of users (L_m) accepts the identification of the user N_k then he sends the following value: $\sigma_j m = S_j(0)H(N_k) = F(0, h(N_j))H(N_k) \in G$.

3) The user N_k should receive t such values to compute the secret key sk_m where

$$sk_m = F(0,0)H(N_k) = sH(N_k) \in G.$$

Then the Encryption and Decryption is done as discussed in [7].

3.5 Subgroup Operations

As mentioned in the initialization phase, each user adopts Shamir secret sharing scheme and holds the shares of secret key of the entire system corresponding to the threshold t. These shares can be used by the users in order to perform certain operations with minimum of t nodes being involved in the network. In our system, the nodes

encrypt the messages among the subgroup(sub) of users by using the Subgroup key. The decryption is possible only when t^1 users in the subgroup cooperate. Now, if a member of the subgroup wants to decrypt the message then the following steps are to be followed in order to get the share of its secret key:

1) The user N_k approaches a group of nodes (L_m) having minimum of t' users.

2) Any user (N_j) in L_m accepting the identity of the new user N_k need to send the following value to N_k:

$$\begin{aligned}\tau_k &= S_j(h(N_k))H(ID_{sub}) \\ &= F(h(N_j), h(N_k))H(ID_{sub}) \in G.\end{aligned}$$

3) The share of the user N_k is computed by using lagrange's interpolation after the user N_k has received t^1 such distinct values (as in above step).The share of the user is given by:

$$[SK_{sub}]_k = F(0, h(L_m))H(ID_{sub}) \in G.$$

3.6 Example

Setup.

- Let the initial set of users $N = \{N_1, N_2, N_3, N_4\}$ No. of users L = 4.
- Public Parmeters: An additive group G of prime order q=4019.
  - The curve used is $E(F_{4019}) : y^2 = x^3 + 1$ k1=67(field of polynomials).
  - The Generator is $P = E(3198, 578)$, Let $th = 3$ and $th^1 = 2$.
- A collision resistant explicit hash function - HTR.
- A collision resistant explicit hash function - HTP.
- Each user chooses a random bivariate polynomial in GF(67):

$$\begin{aligned}N1 &= 3x^2z + 3z^2x + 8xz + 5z + 5x + 2 \\ N2 &= 5x^2z + 5xz^2 + 3xz + 8z + 8x + 5 \\ N3 &= 8x^2z + 8xz^2 + 5xz + 3x + 3z + 3 \\ N4 &= 2x^2z + 2xz^2 + 4xz + 8z + 8x + 4.\end{aligned}$$

- The implicit polynomial defined by all the users is

$$\begin{aligned}F(x, z) &= N_1 + N_2 + N_3 + N_4 \\ &= 18x^2z + 18xz^2 + 20xz + 24x + 24z + 14.\end{aligned}$$

The secret s of the NETWORK is $F(0, 0) = 14$.

- Each user secretly sends to each of other founding users the univariate polynomial $F_{ij} = F_i(x, h(N_j))$.
- The hash values of the users computed using standard hash function are

$$\begin{aligned}h_{n1} &= HTR('user1', k1) = 37 \\ h_{n2} &= HTR('user2', k1) = 54 \\ h_{n3} &= HTR('user3', k1) = 25 \\ h_{n4} &= HTR('user4', k1) = 17\end{aligned}$$

Share Distribution.

- Each user sends the following values to other users:
- N1 also includes $Y_1 = 2Q = (167, 1358)$, $N_{11} = 44x^2 + 53x + 53$, $N_{12} = 28x^2 + 6x + 4$, $N_{13} = 8x^2 + 3x + 60$, $N_{14} = 51x^2 + 3x + 20$. Similarly N2, N3 and N4 also send data to other users.
- Then all the users calculate their secret univariate polynomial from the received values.

$$\begin{aligned}S_1(x) &= 63x^2 + 13x + 31 \\ S_2(x) &= 34x^2 + 59x + 37 \\ S_3(x) &= 48x^2 + 49x + 11 \\ S_4(x) &= 38x^2 + 5x + 20.\end{aligned}$$

- The public key, $PK = sQ = 14E(3198, 578) = E(100, 1874)$.
- PK should also be equal to $Y_1 + Y_2 + Y_3 + Y_4 = E(167, 1358) + E(152, 1437) + E(1356, 3203) + E(3863, 2497) = E(100, 1874)$.

Network Communication Example.

- If user N_5 wants to join the network, It should identify it self to 3 other users and request for acceptance: $\{N_1, N_2, N_3\}$,

$$h_{n5} = HTR('user5', k1) = 27.$$

- N_5 receives the following values

$$N_{15} = 12, N_{25} = 18, N_{35} = 12.$$

- N_5 computes its secret univariate polynomial by using Lagrange interpolation:

$$S_5(x) = 17 * x^2 + 18 * x + 59.$$

Obtention of Individual Keys by Indentity Based Encryption (IBE) Scenario Example.

- Take a public parameter $P_{Pub} = msk \times P$.
- The hash to the point HTP method signature is $QHTP(E, p, q, id, hashfcn)$.
- The key generation is: This method takes public params and secret and generate the key to respective ID.
  $defKeyGen(E, p, q, hashfcn, msk, id)$:

$$\begin{aligned}Q_{id} &= HTP(E, p, q, id, hashfcn) \\ sk_{id} &= msk \times Q_{id} \\ sec &= (Q_{id}, sk_{id})\end{aligned}$$

Return sec.

- Encrypt,This method take public params ,id,message and return cipher text

$$Encrypt(p, q, P, id, m, Q_{id}).$$

- Decrypt,This method takes public params, secret key and cipher text and decrypt the message.

$$Decrypt(p, q, P, C, S_{id}).$$

- now user n2 want to send the message $m = 1712$ to user n1. n2 calls encrypt method.

$$C = Encrypt(p, q, P,' Node1', m, Q_{id1});$$

Cipher text is (1807, 1481) 1718, after receiving encrypted message user n1 calls decrypt method.

$$msg = Decrypt(p, q, P, C, Sk_{id1}).$$

After decrypting message is m= 1712.

Threshold Decryption on Sugroup Example.

- Take the shares of users L={N1,N2,N3,N4} as a subgroup. Each user is having its own secret polynomial.

$$\begin{aligned} S_1(x) &= 63x^2 + 13x + 31; \\ S_2(x) &= 34x^2 + 59x + 37 \\ S_3(x) &= 48x^2 + 49x + 11 \\ S_4(x) &= 38x^2 + 5x + 20. \end{aligned}$$

- create an id for the sub group. $hsg1234 = hfun('SG1234', q)$.
- To find share of jth node remaining users contribute their shares and lagranges interpolation is applied.
  Share of user1 (3125, 1868), user2 (2292, 3913), user3 (2350, 780) and user4 (163, 2657).
  To verify the shares of users caluclate the hash of users they are $rd1, rd2, rd3, rd4$.

$$\begin{aligned} sg1 &= int(rd1)HTP(E, p, q,' SG1234', hashfcn) \\ sg2 &= int(rd2)HTP(E, p, q,' SG1234', hashfcn) \\ sg3 &= int(rd3)HTP(E, p, q,' SG1234', hashfcn) \\ sg4 &= int(rd4)HTP(E, p, q,' SG1234', hashfcn). \end{aligned}$$

  The shares of the users must be (Calculated from F). Share of user1 (3125, 1868), user2 (2292, 3913), user3 (2350, 780) and user4 (163, 2657). Secret of Subgroup is $14HTP(E, p, q,' SG1234', hashfcn)$; Secret of Subgroup is (3857,1351).
- Secret of a subgroup is Lagrange interpolation is applied on users then we get $k1, k2, k3$ from nodes $n1, n2, n3$.

$$\begin{aligned} a1 &= int(k1)sg1; \\ a2 &= int(k2)sg2; \\ a3 &= int(k3)sg3. \end{aligned}$$

  Secret of subgroup $a1 + a2 + a3 = (3857, 1351)$.
- Here ENCRYPTION and DECRYPTION methods are same but in decryption method we have the one more parameter *i.e.* k11 for user 1 it is formed from Lagrange interpolation with t^1 users.
- Any user want to send a message. let $m = 50$, encrypt the message.

$$C = Encrypt(p, q, P,' SG123', m1, Q_{id}).$$

  Threshold is 2 so any two users $n1$ and $n2$ compute l1 = Decrypt(p,q,P,C,sg1,k11); l2 = Decrypt(p,q,P,C,sg2,k22); r1 = l1 l2; and r = HTR(r1,q).
  Now the encrypted message with r output is message = 50.

4 Security Analysis

For a public key encryption scheme the acceptable notion for security is cipher-text security. But the definition concerning the chosen cipher-text should be strengthened. This is because if an adversary outbreaks the public ID of an identity based system then the adversary might posses the private keys of the users. Thus the designed system should withstand such an attack and should be secure. We assume that the identity based encryption system is secure against chosen cipher-text attack.

Note: The adversary A should not have any advantage against the challenger.

Setup: The initialization phase is run by the challenger by taking the security parameter k as input. The system parameters are obtained by the adversary but the master key is kept with it.

Phase 1: The adversary issues either the extraction query or the decryption query.

- Extraction Query: The extract algorithm (defined in 3.1) is run by the challenger. As a result of this, the private key is generated corresponding to particular public key. This private key is sent to the adversary.
- Decryption Query: The extract algorithm (defined in Section 3.1) is run by the challenger. As a result of this, the private key is generated corresponding to particular public key. The decrypt algorithm is run by it using the private key to decrypt the cipher-text. The resulting plain text is sent to the adversary.

Challenge: Two equal length plain texts and ID are generated by adversary after Phase 1 is over. ID is the parameter on which the adversary desired to be challenged.ID did not appear anywhere in the extraction of query in phase 1. A random bit b $\in$ {0,1} is picked by challenger and sets c=Encrypt(parameters,ID,M_b). Challenger sends C to the adversary as a challenge.

Phase 2: In this phase more and more queries are posed by the adversary and it can be either of the following:

- Extraction Query: Here $ID_i \neq$ ID.Then the challenger replies as in phase1.
- Decryption Query: Challenger replies as in phase1 if $(ID_i, C_i) \neq (ID, C)$. Here C is the cipher-text notation.

Guess: $b^1 \in \{0, 1\}$ is displayed by the adversary and the game is won by adversary if $b^1 = b$.

Adversary A has the advantage of attacking the identity based scheme with the help of following function: The function takes the security parameter k as input.Adv[k]=$|Pr[b^1 = b] - \frac{1}{2}|$. This is done by the random bits chosen by the adversary and challenger. The security of the chosen cipher text is demonstrated with the help of this game for Identity based encryption schemes.

Attack: Let the number of players trying to recover the secret S_i be less than or equal to $t_i - 1$. Here t is the threshold value.

Analysis: The recovery of the secret in the proposed scheme completely revolves around the concept of Lagrange's Interpolation polynomial. In order to solve t_i in the process of getting to know the unknown symbol, we are definitely going to need t_i number of equations. Therefore, it is only t_i or more players who can have a complete knowledge of the secret. There is no chance for t_i or lesser players to crack the secret.

5 Conclusion

Now a days many applications demands the network without the presence of trusted third party (TTP). This can be achieved by distributing the role of TTP among the network users using secret sharing concept. In our paper we proposed an efficient way to decentralize the network and to establish a secure communication among the users of the network using Identity based encryption. We also discussed the suitable protocol to perform sub group operations among the sub set of users of a network. Our scheme is useful for the applications where secure communication is required without the presence of trusted third party.

References

[1] J. Anzai, N. Matsuzaki and T. Matsumoto, "A quick group key distribution scheme with entity revocation," in *International Conference on the Theory and Application of Cryptology and Information Security*, pp. 333-347, 1999.

[2] A. Basit, N. C. Kumar, V. C. Venkaiah, S. A. Moiz, A. N. Tentu and W. Naik, "Multi-stage Multi-secret sharing scheme for hierarchical access structure," in *International Conference on Computing, Communication and Automation (ICCCA'17)*, pp. 557-563, 2017.

[3] G. R. Blakley, "Safeguarding cryptographic keys," in *Proceedings of the National Computer Conference*, vol. 48, pp. 313-317, 1979.

[4] C. Blundo, A. D. Santis, A. Herzberg, S. Kutten, U. Vaccaro and M. Yung, "Perfectly-secure key distribution for dynamic conferences," in *Annual International Cryptology Conference*, pp. 471-486, 1992.

[5] D. Boneh and M. Franklin, "Identity-based encryption from the weil pairing," in *Annual International Cryptology Conference*, pp. 213-229, 2001.

[6] V. Daza, J. Herranz and G. Sez, "Constructing general dynamic group key distribution schemes with decentralized user join," in *Australasian Conference on Information Security and Privacy*, pp. 464-475, 2003.

[7] V. Daza, J. Herranz, P. Morillo and C. Rafols, "Cryptographic techniques for mobile ad-hoc networks," *Computer Networks 51*, no. 18, 2007.

[8] G. Frey, M. Muller and H. G. Ruck. "The Tate pairing and the discrete logarithm applied to elliptic curve cryptosystems," *IEEE Transactions on Information Theory 45*, no. 5, pp. 1717-1719, 1999.

[9] O. Goldreich, R. Ostrovsky, E. Petrank, "Computational complexity and knowledge complexity," vol. 27, no. 4, pp. 1116-1141, 2001.

[10] M. S. Hwang, C. C. Lee, J. Z. Lee, and C. C. Yang, "A secure protocol for bluetooth piconets using elliptic curve cryptography", *Telecommunication Systems*, vol. 29, no. 3, pp. 165-180, 2005.

[11] A. Joux, "A one round protocol for tripartite Diffie-Hellman," in *International Algorithmic Number Theory Symposium*, pp. 385-393, 2000.

[12] A. Khalili, J. Katz and W. A. Arbaugh, "Toward secure key distribution in truly ad-hoc networks," in *Symposium on Proceedings of Applications and the Internet Workshops*, pp. 342-346, 2003.

[13] N. Koblitz, "Elliptic curve cryptosystems," *Mathematics of Computation 48*, no. 177, pp. 203-209, 1987.

[14] N. C. Kumar, A. Basit, P. Singh, V. C. Venkaiah and Y. V. Rao, "Node authentication using BLS signature in distributed PKI based MANETS," *Cryptography and Security*, vol.9, no. 4, 2017.

[15] N. C. Kumar, A. Basit, P. Singh and V. C. Venkaiah "Proactive secret sharing for long lived MANETs using elliptic curve cryptography," in *IEEE International Conference on Inventive Computing and Informatics (ICICI'17)*, pp. 312-316, 2017.

[16] H. Luo, J. Kong, P. Zerfos, S. Lu and L. Zhang, "URSA: Ubiquitous and robust access control for mobile ad hoc networks," *IEEE/ACM Transactions on Networking (ToN'04) 12*, no. 6, pp. 1049-1063, 2004.

[17] A. J. Menezes, T. Okamoto and S. A. Vanstone, "Reducing elliptic curve logarithms to logarithms in a finite field," *IEEE Transactions on Information Theory 39*, no. 5, pp. 1639-1646, 1993.
[18] J. Pan, L. Cai, X. S. Shen and J. W. Mark, "Identity-based secure collaboration in wireless ad hoc networks," *Computer Networks 51*, no. 3, pp. 853-865, 2007.
[19] R. Sakai, K. Ohgishi and M. Kasahara, *Cryptosystems Based on Pairing*, SCIS 2000-C20, Jan. 2000. (`https://www.researchgate.net/publication/243538884_Cryptosystem_based_on_Pairings`)
[20] A. Shamir, "Identity-based cryptosystems and signature schemes," in *Workshop on the Theory and Application of Cryptographic Techniques*, pp. 47-53, 1984.
[21] A. Shamir, "How to share a secret," *Communications of the ACM 22*, no. 11, pp. 612-613, 1979.
[22] A. Shamir, "Identity-based cryptosystems and signature schemes," in *Workshop on the Theory and Application of Cryptographic Techniques*, pp. 47-53, 1984.
[23] N. Singh, A. N. Tentu, A. Basit and V. C. Venkaiah, "Sequential secret sharing scheme based on Chinese remainder theorem," in *IEEE International Conference on Computational Intelligence and Computing Research (ICCIC'16)*, pp. 1-6, 2016.
[24] A. N. Tentu, A. Basit, K. Bhavani and V. C. Venkaiah, "Multi-secret sharing scheme for level-ordered access structures," in *International Conference on Number-Theoretic Methods in Cryptology*, pp. 267-278, 2017.
[25] X. Yi, "An identity-based signature scheme from the Weil pairing," *IEEE Communications Letters 7*, no. 2, pp. 76-78, 2003.
[26] L. Zhou and J. H. Zygmunt, "Securing ad hoc networks," *IEEE Network 13*, no. 6, pp. 24-30, 1999.

Biography

N Chaitanya Kumar received M.Tech from JNTU Hyderabad and he did Bachelor degree in computer science. Currently, he is pursuing his PhD in Computer Science from the University of Hyderabad. His research interests include Information security, Cryptography in MANET.

Abdul Basit received Master of computer application from Jamia Hamdard University New Delhi. He did Bachelor of Science in Information technology from SMU Gangtok. Currently, he is pursuing his PhD in Computer Science from the University of Hyderabad. His research interests include Information security, Cryptography and Cyber security.

Priyadarshi Singh received M.Tech from IIT(ISM) Dhanbad. He did Bachelor degree in Information Technology. Currently, he is pursuing his PhD in Computer Science from the University of Hyderabad. His research interests include Cryptography, Public key infrastructure.

V. Ch. Venkaiah obtained his PhD in 1988 from the Indian Institute of Science (IISc), Bangalore in the area of scientific computing. He worked for several organisations including the Central Research Laboratory of Bharat Electronics, Tata Elxsi India Pvt. Ltd., Motorola India Electronics Limited, all in Bangalore. He then moved onto academics and served in IIT Delhi, IIIT Hyderabad and C R Rao Advanced Institute of Mathematics, Statistics, and Computer Science. He is currently serving in School of Computer and Information Sciences, University of Hyderabad. He is a vivid researcher. He designed algorithms for linear programming, subspace rotation and direction of arrival estimation, graph coloring, matrix symmetriser, integer factorisation, cryptography, knapsack problem, *etc.*

A *k*-anonymous Location Privacy Protection Method of Dummy Based on Geographical Semantics

Yong-Bing Zhang[1,2], Qiu-Yu Zhang[1], Zong-Yi Li[2], Yan Yan[1], and Mo-Yi Zhang[1]
(Corresponding author: Qiu-Yu Zhang)

School of Computer and Communication, Lanzhou University of Technology[1]
No. 287, Lan-Gong-Ping Road, Lanzhou 730050, China
(Email: zhangqylz@163.com)
Gansu Institute of Mechanical & Electrical Engineering[2]
No. 107, Chi-Yu Road, Tianshui, Gansu 741001, China
(Received Aug. 2, 2018; Revised and Accepted Feb. 7, 2019; First Online June 17, 2019)

Abstract

Dummy is one of the main methods used to protect location privacy. In existing methods, the efficiency of dummy generation is low, and the geographical semantic information of location is not fully taken into account. In order to solve these problems, a *k*-anonymous location privacy protection method of dummy based on geographical semantics was proposed in this paper. Firstly, the location data set in the rectangle region containing the real location is obtained from WiFi APs. Secondly, adopting the multicenter clustering algorithm based on max-min distance, some locations are selected. Its geographical distance between them is the farthest, and a candidate set of dummies is generated. Finally, by calculating the edit-distance between geographic name's information of locations, the semantic similarity between any two locations in the candidate set is obtained, and *k*-1 locations with the minimum semantic similarity are selected as dummies. Experimental results show that the proposed method can ensure the physical dispersion and semantic diversity of locations, as well as the improvement of the efficiency of dummy generation. Meanwhile, the balance between privacy protection security and query service quality is achieved.

Keywords: Clustering Center; Dummy; Geographic Semantics; k-*anonymous; Location Privacy Protection; Semantic Similarity*

1 Introduction

With the development of mobile location technology and wireless communication technology, a large number of mobile devices in the market have capacity of GPS precise positioning, which makes location-based service (LBS) become one of the most promising services to mobile users [33]. However, when LBS provide convenience and great benefits to the society, its problem of sensitive information leakage has attached more attentions by many people. Because user's location is shared among different location service providers (LSPs), untrustworthy third parties can easily steal user's privacy via analyzing and comparing these locations' information [35]. For example, through capturing recent users' trace, some information can be analyzed by adversary such as home addresses, workplaces, and health conditions, etc. Therefore, it is necessary to ensure the safety of users' location privacy.

Currently, in order to prevent the leakage of privacy information, many different methods are proposed by experts and scholars, including fuzzy method, encryption method and strategy-based method. Because of the better reliability, the fuzzy method is the most commonly used in the field of location privacy protection, which is mainly realized by means of spatial anonymity or dummy technology. The spatial anonymous method usually needs the help of Fully-Trusted Third Party (TTP) [16]. When a location query service is needed, the mobile user first sends the query request to the TTP, a *k*-anonymous region containing the user's location is generated by the TTP and then it will be sent to the LBS server for query. In this method, if the area of *k*-anonymous region is too large, it not only consumes more time, but also reduces the accuracy of the query result. Meanwhile, TTP is easy to become a bottleneck of system. However, in the dummy-based location privacy protection, TTP and anonymous region are not required, and the dummy locations are generated by mobile clients. Thus, it can compensate the above disadvantages of spatial anonymous methods well.

In the dummy-based location privacy protection, in or-

der to improve the efficiency of dummy location generation and the query service quality, a k-anonymous location privacy protection method of dummy based on geographical semantics was proposed. In this paper, we give full consideration to the geographical semantic information features of locations. Firstly, adopting multi-center clustering algorithm based on max-min distance (MCAMD) [24], a number of cluster centers are generated by clustering calculation, which constitute a candidate set of dummies. Then, using edit-distance [37] to calculate semantic similarity of geographic name's information among elements in candidate set, and k-1 locations with the minimum semantic similarity are selected as dummies. The proposed method can meet semantic l-diversity and physical dispersion of locations, and improve the efficiency of dummies generation. Furthermore, it improves the query service quality.

Our main contributions can be summarized as follows:

1) A dummy selection method considering the geographical semantic information characteristics of locations is proposed, which balanced the contradiction between privacy protection and query quality.

2) A multi-center clustering algorithm based on the max-min distance method is used to generate candidate set of dummies, which can ensure the physical dispersion of the dummies.

3) We calculate the semantic similarity between geographic name's information of locations, and the locations with the smallest semantic similarity is selected as the dummies, which ensures the semantic diversity of the dummies.

The remaining part of this paper is organized as follows. Section 2 reviews related work of location privacy protection. Section 3 gives system model of this paper. Section 4 describes two algorithms and analysis. Section 5 gives the experimental results and performance analysis as compared with other related methods. Finally, we conclude our paper in Section 6.

2 Related Work

The location privacy protection method is divided into two main categories according to the system architecture, including distributed structure [21]based on Peer-to-Peer (P2P) [23]and central server structure based on TTP [29]. In the distributed structure, location privacy protection is accomplished through collaboration between users. Chow *et al.* [4, 6, 7] proposed a P2P-based spatial anonymity method. In these methods, the k-anonymous privacy protection based on distributed architecture is achieved by using location information of neighbors' node, but the security of the neighbors' node is ignored. The P2P-based scheme is simple and flexible, but which greatly increases various overhead of the smart phone. Furthermore, users are mobile rather than static [34]. In a centralized structure based on TTP, a method of location privacy protection based on TTP is proposed by Zhou *et al.* [36]. This structure mode has good effect of privacy protection, but TTP also needs to be protected. Li *et al.* [12] proposed a location privacy protection scheme based on efficient information cache, which reduces the number of times that the users' access to TTP, the query efficiency is improved, and the probability of information leakage is reduced, but the burden of the mobile client is increased.

In addition, Cheng *et al.* [3] put forward an independent structure model, and users protect location privacy according to their own abilities and knowledge. The structure of this method is simple, which is easy to merge with other structures, but it requires high performance for mobile clients. Li *et al.* [11] put forward a multi-server architecture, users can be divided into different subsets according to the security requirements, and each location server can only obtain partial subset. The concealment of location is improved in this method, but it is mainly suitable for the social network. Mouratidis *et al.* [15] put forward a location privacy protection method based on privacy information retrieval, and its location privacy protection is implemented by using retrieval and encryption. The location privacy is well protected in this method, but it increases the overhead of communication and hardware, and reduces the query service quality. With the maturity and popularity of cloud service technology, Kim *et al.* [10] proposed a location privacy protection method based on searchable encryption. By accessing to the cloud server in the encrypted state, the security of location data and query records is guaranteed, but query efficiency and query accuracy need to be improved further.

In the recent researches, k-anonymity [25] is still the mainstream method of location privacy protection, which was born in the relational database, and its key attribute is dealt with using generalization and fuzzy technology. So none of the records can be distinguished from other k-1 records, and the location anonymity is realized. The method of k-anonymity location privacy protection is mainly divided into spatial region anonymity and dummy anonymity. Gruteser *et al.* [5] proposed a k-anonymity location privacy protection method, and its location privacy is protected by constructing k-anonymous region. The region must meet two conditions: 1) The area of the region reaches a certain value; 2) There are k users in the region. Due to the above tow limitations, the effect of location privacy protection is improved, but all users must have the same location anonymity requirement. Bamba *et al.* [1] put forward a method of grid partition, which provide two algorithms: Top-Down Grid Cloaking algorithm and Bottom-Up Grid algorithm, which can be selected according to the users' needs. Xu *et al.* [27] proved that the size of k-anonymous region has a great impact on the accuracy of query results, which provides guidance for the research of anonymous region partition. On this basis, some anonymous region construction methods with various geometric shapes were presented in [20, 26, 30, 31].

However, these methods have two serious shortcomings: First, it must rely on TTP, but TTP is not absolutely secure, and it's easy to become the bottleneck of the system. Second, the size of anonymous region and the accuracy of query results are a pair of contradiction, and the larger the anonymous region, the better the effect of privacy protection, but the accuracy of the query results will be reduced.

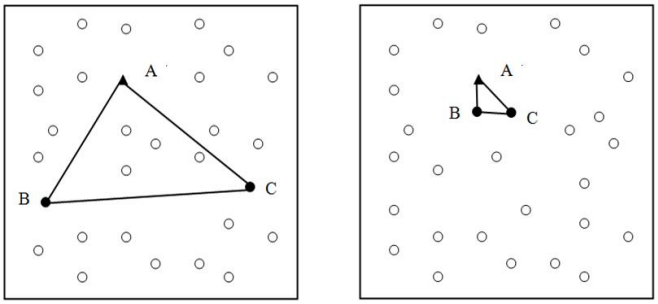
Because of these above serious shortcomings in the spatial anonymity method, the k-anonymity method of dummy has been widely used. The dummy method was first introduced into the location privacy protection by Kido *et al.* [8, 9] in 2005. And then Lu *et al.* [14] proposed a method of randomly adding dummy locations in a circular or rectangular region, users can select dummy locations in the region according to their demands. Several dummy-based privacy protection methods for continuous queries in mobile trajectories were proposed in document [13,28,32]. Niu *et al.* [18] took into account the adversaries' attack with background information, and DLS algorithm and improved DLS algorithm were proposed. Then, Niu *et al.* [19] introduced cache mechanism into the location privacy protection, a cache-based dummy selection algorithm (CaDSA) was proposed, which has improved the query efficiency. Next, Niu *et al.* [17] proposed a mobile location privacy protection scheme named DUMMY-T, which aims to protect user's location privacy from background attacks, the dummy is generated by the dummy location generation (DLG) algorithm, and dummy path is generated by the dummy path construction (DPC) algorithm, which ensure the security of location privacy. Sun *et al.* [22] selected dummy locations by probability estimation, which can prevent probability attack, and solve the problem that attackers can judge the real location information by analyzing historical records.

3 System Model

3.1 Attack Model

In location-based services, common attacks include background attack, probability attack and semantic attack. For background attacks, location privacy is usually protected by eliminating the link between background information and the user's current location. In general, in order to overcome the probability attack, after obtaining the history query record of the query user, the locations with high query probability is used as the dummies to confuse the attacker. However, there are many forms of semantic attack, so the difficulty of protection is large.

In existing study, most methods select dummy according to query probability, which seldom consider the location's geographical semantic information, and adversaries can easily obtain user' location by analyzing the geographical semantic information. As shown in Figure 1, solid triangle A represents real position, Hollow circle represents dummy candidate set, and solid circle B and C represent the selected dummy locations.



(a) (b)

Figure 1: A sample of the location similarity attack: (a) Semantic features; (b) Geographical features

As shown in Figure 1(a), A, B and C are the selected dummy locations, assuming that the three locations are in the hospital, and adversaries can easily identify that users have health problems through semantic analysis. The selected dummies are too close to the real location in Figure 1(b), adversaries can easily find the exact location of the user by computing geographical distance. Therefore, the selection of dummies should consider the geographical semantic information of the location as much as possible, which can ensure the physical dispersion and semantic diversity of all locations including the real location, and further improve the effect of location privacy protection.

In order to prevent location privacy leakage due to geographical semantic attacks, Chen *et al.* [2] proposed a dummy selection method based on semantic-aware. The physical dispersion and semantic diversity of dummies are guaranteed. However, this method needs to repeatedly calculate the physical distance and semantic distance between all locations, and the efficiency is relatively low when location data is large. Furthermore, it needs construct semantic tree for locations in WiFi APs to compute the semantic distance, the burden of WiFi APs and the time of preprocessing is increased, and the service quality is reduced.

3.2 System Structure

In the TTP-based central server model, if users initiate more queries, TTP is easy to become a system bottleneck. Furthermore, TTP is not absolutely safe and reliable. Once TTP is attacked, all locations privacy will be leaked. So, we use a system model without TTP in this paper, and the generation of dummies and the sending of query requests are all accomplished by the mobile client. The system structure model is shown in Figure 2.

In this system structure, the mobile user obtains the location information of the region including the real location from WiFi APs, as shown in Figure 3 and Figure 4. Firstly, by adopting the MCAMD algorithm, a number of cluster centers are generated in mobile client. These locations are the farthest from each other, the dummy can-

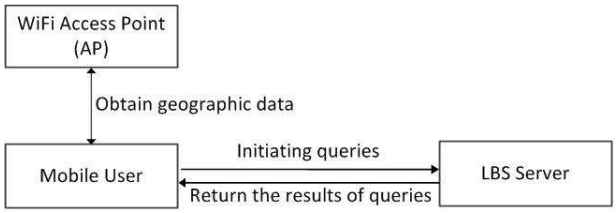


Figure 2: System structure model

didate set is generated from them. Then, the semantic similarity is calculated for the location information of the candidate set, and the k-1 locations with the minimum semantic similarity are selected as the dummies. Finally, the mobile user sends k-1 dummies and real location to the LBS server to query.

Figure 3: Selected location region

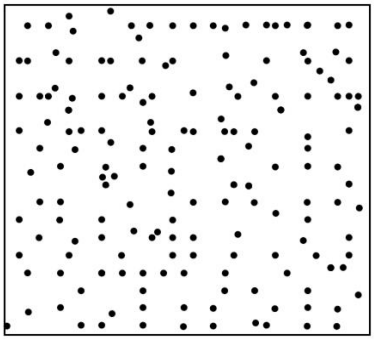


Figure 4: Geographical location in the region

3.3 Definition

Definition 1. *Let R_s represents the selected rectangular area, and $S_n = \{l_1, l_2, \cdots, l_n\}$ represents the set of locations in the rectangle region.*

Definition 2. *Let l_{phi} represents the physical distance between any two locations, and l_{sem} represents the set of locations in the rectangle region.*

Definition 3. *Let $S_1 = \{l_1, l_2, \cdots, l_m\}$ represents the candidate set that satisfies physical dispersion, and $S_2 = \{l_1, l_2, \cdots, l_{k-1}\}$ represents the dummy set that satisfies semantic diversity. Let l_{real} represents the user's location, and the location result set includes the dummy set S_2 and real location l_{real}.*

Definition 4. *If the semantic similarity between l_i and l_j satisfies the following conditions: $1 - \frac{|SEM_{t_i}|}{C_k^2} \geq \theta$, where $SEM_{t_i} = \{l_{sem} | l_{sem}(l_i, l_j) \leq l\}$, k=$|RS_{t_i}|$, C_k^2 is a combination formulas, l is the default threshold of the semantic diversity. Then, the result set RS_{t_i} is called a θ-security set, and the purpose of privacy protection is to get the maximum value of θ by 1, the semantic similarity between l_i and l_j is equal or less than 0.2.*

4 Algorithmic Description

The proposed method of dummy generation is implemented by the following two algorithms: The dummy candidate set S_1 is generated through cluster calculation in Algorithm 1. In Algorithm 2, the dummy set S_2 is generated by calculating semantic similarity of locations in the candidate set S_1.

4.1 Algorithm 1

Algorithm 1: Calculating physical distance and obtaining dummy locations set.

Input: Location data set S_n, demand parameter m.

Output: Generate a dummy candidate set S_1.

Step 1: Given γ value, $0 < \gamma < 1$.

Step 2: The real location l_{real} is taken as the first cluster center Z_1.

Step 3: Find the location that it is the farthest location from Z_1, which is treated as a second Cluster center Z_2.

Step 4: For each l_i of the remaining objects in S_n, its distance to Z_1 and Z_2 is D_{i1} and D_{i2}. Assumed that D_{12} is the distance between Z_1 and Z_2, if $D_i = max\{min(D_{i1}, D_{i2})\}(i = 1, 2, \cdots, n)$ and $D_i > \gamma \cdot D_{12}$, then take l_i as the third Cluster center Z_3.

Step 5: And so on, get all the v clustering centers that conforms to the conditions. When max-min distance is lower than $\gamma \cdot D_{12}$, the calculation for finding the cluster center is finished.

Step 6: Suppose v represents the number of cluster centers obtained by calculation, judge:

1) If $v \geq m$, the algorithm is over, then Step 7,
2) If $v < m$, re-select the γ value, and turn to Step 1.

Step 7: The dummy candidate set S_1 is generated.

4.2 Algorithm 2

Algorithm 2: Calculating semantic similarity and obtaining dummy location result set.

Input: Location candidate set S_1, semantic diversity parameter threshold l.

Output: Location result set S_2.

Step 1: Matching each character of the place name information in turn, and ignore the same prefix characters with the same matching values. Then, get two new character strings A and B.

Step 2: Suppose that the string A contains i characters and it is represented as A=$a_1a_2a_3La_i$; the string B contains j characters and it is represented as B=$b_1b_2b_3Lb_j$.

Step 3: A dynamic programming matrix of i+1 columns and j+1 rows is constructed. The last element obtained from $D[i, j]$ is $ed(A,B)$.

Step 4: If j=0, return i and then exit; if i=0, return j and then exit.

Step 5: The first row is initialized to 0,1,L,i; the first column is initialized to 0,1,L,j.

Step 6: Assign values for each element in the matrix: if a_i=b_i, then $D[i, j]$=$D[i$-1, j-1]; if a_i≠b_i, then $D[i, j]$=1+min($D[i$-1, j-1], $D[i$-1, $j]$, $D[i, j$-1]).

Step 7: Repeat step 6, until all the values in the matrix are obtained, the final edit-distance is $D[i, j]$.

Step 8: Calculating similarity matching index $S(A, B)$ through $D[i, j]$, that is Semantic similarity.

Step 9: Select the k-1 locations with the minimum semantic similarity, and dummy result set S_2 is generated.

4.3 Algorithm 1 Description

Using the MCAMD algorithm to compute cluster center for location geographic coordinates in a square region, several cluster centers are obtained, which are selected as dummy candidate set. The MCAMD algorithm is a clustering algorithm based on heuristic, which takes as far away objects as cluster centers according to Euclidean distance. Firstly, a sample object is used as the first cluster center, and then a sample which is the farthest from the first cluster center is selected as the second cluster center. Then determine the other cluster centers, until there is not new cluster center. After determining all the clustering centers, the clustering sample set including m samples is taken as dummy location candidate set.

The example of cluster center calculation is shown in Figure 5, there are ten locations in the region. According to Algorithm 1, l_1 is selected as first cluster center, and then l_5 is selected as second cluster center, and then third cluster center l_9 is determined. After clustering calculation, three clustering centers are obtained, and the dummy location candidate set is generated.

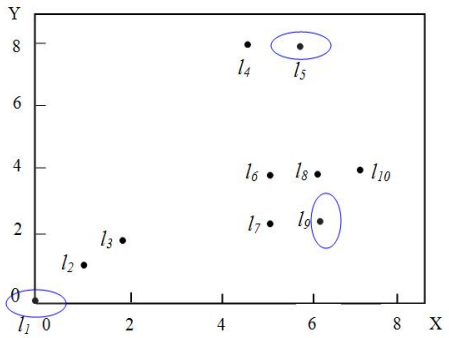


Figure 5: Example of MCAMDA algorithm

When determining the cluster center, the real location is used as the initial cluster center Z_1. If l_i is selected as the i-th clustering center, the conditions must be satisfied.

$$D_i > \gamma \cdot D_{12}(i = 1, 2, \cdots, n) \tag{1}$$

where $D_i = max\{min(D_{i1}, D_{i2})\}(i = 1, 2, \cdots, n)$, $D_{12} = |Z_2 - Z_1|$, γ is the test parameter in the algorithm, it's usual value is $0.5 \leq \gamma < 1$.

4.4 Algorithm 2 Description

Algorithm 2 computes semantic similarity for location information of dummy candidate set. Firstly, according to the characteristics of Chinese geographical names, the same prefix in place name information is eliminated. Then, by calculating semantic similarity for the remaining string of place name through the edit-distance, the efficiency and the accuracy of calculation is improved. For example, "Guangzhou second middle school" and "Guangzhou Tie Yi middle school" are two strings of Chinese place name. The characters of "Guangzhou" do not have any meaning for the calculation of semantic similarity in the two place name strings, and which also affect the accuracy of the calculation results. So "Guangzhou" is eliminated as a prefix in the calculation.

$D[i, j]$ is the edit-distance of the dynamic programming matrix, and the cost of the each edit operation is between 0 and 1. And it can be set different values according to the requirements. In this paper, the value is set to 0 or 1. if $a_i = b_i$, the cost of replacement is 0. Otherwise, the cost of all edit operations is 1. In Equation (2), D is a dynamic programming matrix, which represents the edit-distance between string A= "Second middle school" and string B= "Tie Yi middle school".

$$D = \begin{vmatrix} 0 & 1 & 2 & 3 & 4 \\ 1 & 1 & 2 & 3 & 4 \\ 2 & 2 & 2 & 3 & 4 \\ 3 & 3 & 3 & 2 & 3 \\ 4 & 4 & 4 & 3 & 2 \end{vmatrix} \tag{2}$$

The edit-distance between the two strings is obtained by calculating, which is $D[i, j]$=$D[4, 4]$=2. Using Equation (3) to calculate the similarity matching index between the strings, that is semantic similarity. The semantic similarity is 0.5.

$$S(A, B) = 1 - \frac{D[i, j]}{max\{|A|, |B|\}} \quad (3)$$

Where $|A|$ and $|B|$ represents the length of two strings respectively, and the maximum length of string S is used to calculate semantic similarity.

At last, according to the Equation (4), the k locations with the minimum semantic similarity including the real location are obtained.

$$Arg\,min(S(l_i, l_j)) \quad (4)$$

4.5 Algorithm Analysis

In this paper, firstly, adopting the MCAMD Algorithm, to select the $m(m > 2k)$ locations with the maximum distance from each other as dummies, dummy candidate set including the real location is generated. Then, by calculating the semantic similarity of the locations in the candidate set, the k locations including the real location are selected as the result set. So physical dispersion and semantic diversity of k locations are ensured.

In Algorithm 1, the candidate set of dummies is generated by clustering calculation, and the physical dispersion between different locations is guaranteed. The clustering results of the algorithm are related to the selection of parameter γ and the first cluster center Z_1, and the real location is used as the initial cluster center. To make sure that the numbers of samples in the candidate set is enough, the number of cluster centers m satisfy the condition of $m > 2k$. In this paper, the initial parameter value of γ is 0.5.

In Algorithm 2, the semantic similarity of geographic name's information is obtained by calculating the edit-distance. In the calculation, the more similar the character in the two strings are, the smaller the edit-distance is, while the greater the semantic similarity is. When the two strings are exactly the same, the edit-distance is 0, and the semantic similarity is 1.

5 Experimental Results and Analysis

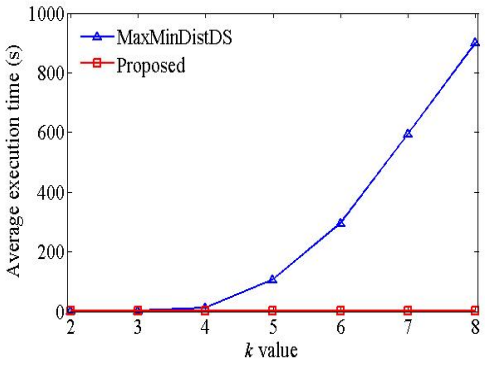
In order to evaluate the performance of the proposed method, we use a real map of Guangzhou from Google maps, and select the 55 WiFi APs in the 8km×8km. The hardware environment of the experiment is as follows: 3.2 GHz Intel Core i5 processor with memory size of 4 GB. The operating system is Windows 7. The proposed method is implemented by Eclipse development platform and Java programming language.

Table 1 is configured for the default parameters of the experiment.

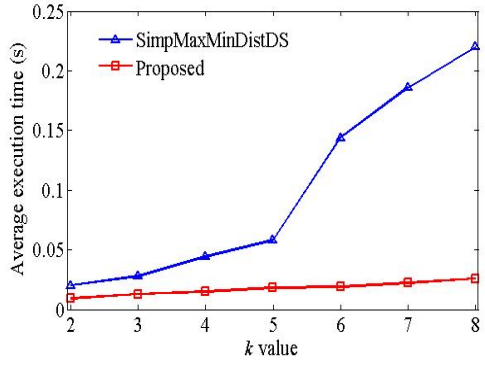
5.1 Average Execution Time

Firstly, the efficiency of the proposed method is verified through experiment. In dummy location selection method considering semantic similarity, we compare the average execution time of dummy locations with MaxMinDistDS [2], SimpMaxMinDistDS [2] and the proposed method. The average execution time of dummy locations of the three methods is shown in Table 2.

In Figure 6, we compare the efficiency of generating dummies with MaxMinDistDS, SimpMaxMinDistDS and the proposed method. As the Figure 6(a) shown, with the increase of k, the MaxMinDistDS algorithm takes much more time than the proposed method. As the Figure 6(b) shown, when $k < 5$, the average execution time of SimpMaxMinDistDS algorithm is slightly larger than that of the proposed method, when $k \geq 5$, the average execution time of SimpMaxMinDistDS algorithm is much larger than that of the proposed method. As can be seen from Figure 6, with the increase of k, the efficiency of the proposed method is more and more advantageous than the other two algorithms.



(a)



(b)

Figure 6: Average generation time of dummy: (a) Comparison between MaxMinDistDS and the proposed; (b) Comparison between SimpMaxMinDistDS and the proposed

In addition, we compare the efficiency of generating dummies with Random [9], Rotation [32], Footprint [28] and DUMMY-T [17] algorithm, as shown in Figure 7.

As can be seen from Figure 7, with the increase of k,

Table 1: Experimental default parameter configuration

Parameter	Value
k	[2, 16]
l	$\leq$0.2
γ	16km×16km
Location set	10000
Space range (km^2)	8×8
WiFi APs Coverage range(m)	800

Table 2: Average execution time vs. k

k	2	3	4	5	6	7	8
MaxMinDistDS	0.07s	1.69s	13s	106.27s	295.41s	592.91s	899.45s
SimpMaxMinDistDS	0.02s	0.028s	0.044s	0.058s	0.0144s	0.186s	0.22s
Proposed	0.009s	0.013s	0.015s	0.018s	0.019s	0.022s	0.026s

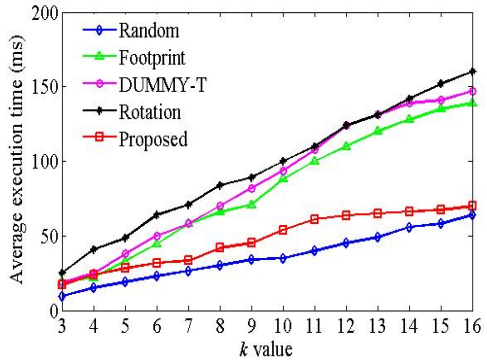


Figure 7: Average generation time of dummy

the average execution time of these algorithms is all increasing. Among them, the average execution time of the proposed and Random algorithm is smaller than other three algorithms. The average execution time of Random algorithm is the least, and the average execution time of the Rotation algorithm is the most. As the Figure 7 shown, when $k\leq4$, the average execution time of Footprint, DUMMY-T and the proposed method are the same. When $k > 4$, The difference in the execution time of the five algorithms is getting bigger and bigger. With the increase of k, average dummy generation time of the proposed method is larger than that of Random algorithm, but it is smaller than the other three algorithms.

Through the analysis of efficiency comparison experiments, it is found that the efficiency of the proposed method is higher than the other three algorithms except Random algorithm. The Random algorithm is randomization, and the effect of privacy protection is relatively poor. The experimental results show that, when the anonymity is large, the proposed method is more efficient under the condition of maintaining good location privacy. Therefore, the efficiency of dummy generation is further improved. And the larger the k is, the better the effect is.

5.2 Comparison of Physical Dispersion

In this paper, we compare the minimum distance of dummies with SimpMaxMinDistDS, MaxMinDistDS and the proposed method. The result is shown in Figure 8.

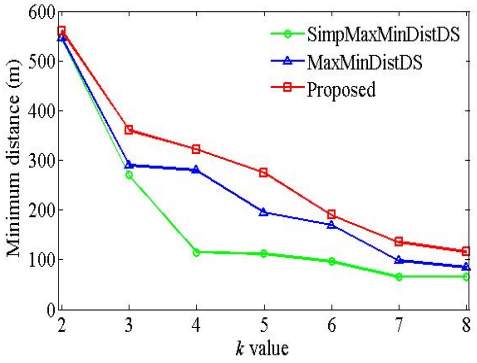


Figure 8: The minimum distance vs. k

As can be seen from Figure 8, the minimum distance between dummies of several methods is mostly reduced with the increase of k, but the minimum distance of the proposed method is obviously larger than the MaxMinDistDS and SimpMaxMinDistDS. Because the proposed method uses the clustering center algorithm in Algorithm 1, and selects the dummies with relatively large distance as the dummy location set, which gives priority to ensuring physical dispersion between locations. However, the MaxMinDistDS first satisfies the semantic diversity and then guarantees the physical dispersion, and the SimpMaxMinDistDS selects the larger in the physical distance and the semantic distance as the dummy location result set. From the experimental result we can see that the proposed method has better physical dispersion.

5.3 Comparison of Semantic Diversity

We compare the semantic diversity with the proposed method, MaxMinDistDS, SimpMaxMinDistDS and DLS [18] through experiment. According to the semantic diversity of the locations in the dummies result set, the θ-secure is obtained, as shown in Figure 9.

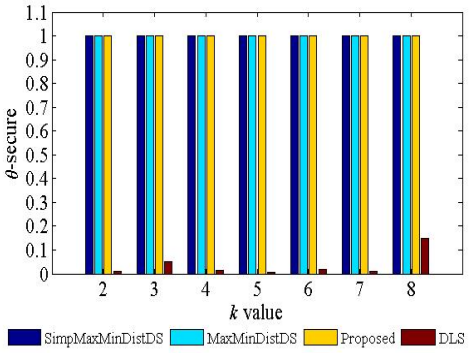


Figure 9: θ-secure vs. k

As can be seen from Figure 9, with the increase of k value, the θ value of algorithm MaxMinDistDS and SimpMaxMinDistDS basically do not change, and always reach 1. The θ value of the proposed method is always the maximum value 1, which can satisfy the requirement of semantic l-diversity. The θ value of DLS is relatively small, and always keeps a lower value. This is because the semantic diversity of geographic location information is considered in MaxMinDistDS, SimpMaxMinDistDS and the proposed method, but in the DLS, the query probability between dummy locations is only considered, and does not consider the semantic information. Moreover, the locations with larger query probability are often in hot areas, and the semantic information between these locations is more similar, thus having greater semantic similarity. Therefore, the semantic diversity of the DLS method is poor, and the θ value is very small.

Through the comparison of experiments, it is found that the proposed method takes less time to generate dummy than other methods. Therefore, the proposed method improves the efficiency of dummy generation, and it further improves the quality of query service. Moreover, through the comparison of experiments, it is found that the physical dispersion and semantic diversity of the dummies selected by this method are better, which can effectively prevent the attack of the opponent who has mastered the characteristics of the geographical semantic information. Therefore, this method not only guarantees location privacy, but also improves the quality of query services, and effectively balanced the contradiction between the effect of location privacy protection and the quality of query service.

5.4 Safety Analysis

In dummy privacy protection, if k dummies are located in one or some areas of concentration, the real location can be easily obtained by reducing the search range. In this case, k-anonymity only meets the requirement in quantity, but does not achieve the effect of anonymity. In the proposed method, the dummies are generated by Algorithm 1, which are distributed uniformly in the region. Therefore, the probability that any location can be distinguished from other k-1 locations is $1/k$, and the effect of anonymity is satisfied. On the basis of geographical distribution, the better the physical dispersion between locations, the better the anonymity.

In semantic attacks, an adversary easily deduces the privacy information of the query user according to the analysis of the semantic relationship between dummies. The greater the difference of the semantic information of geographic name, the better the diversification of location semantics. In Algorithm 2, k locations with the minimum semantic similarity as dummies, it satisfies the requirement of geographic semantics l-diversification.

In conclusion, the proposed method meets the requirements of k-anonymity and l-diversity, and can effectively protect location privacy.

6 Conclusions

In this paper, the issues about location privacy protection based on dummies are discussed, and a k-anonymous privacy protection method of dummy based on geographical semantics was presented. Two algorithms are included in this method: adopting multicenter clustering algorithm based on max-min distance, a number of cluster centers are generated, which constitute a candidate set of dummies in Algorithm 1. In Algorithm 2, by calculating the edit-distance between geographic name's information of locations, the semantic similarity between any two locations in the candidate set is obtained, and dummy location result set with k-1 dummies are generated. We evaluate our algorithms through a series of simulations, which show that our algorithms can ensure the physical dispersion and semantic diversity of locations, protect location privacy effectively, and reduce the time of generating dummy.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (No. 61363078, 61762059), the Research Project in Universities of Education Department of Gansu Province of China (No. 2017B-16, 2018A-187), the Open Project Program of the National Laboratory of Pattern Recognition (NLPR)(No. 201700005). The authors also gratefully acknowledge the helpful comments and suggestions of the reviewers, which have improved the presentation.

References

[1] B. Bamba, L. Liu, P. Pesti, and T. Wang, "Supporting anonymous location queries in mobile environments with privacy grid," in *Proceedings of the 17th International World Wide Web Conference*, pp. 237–246, Jan. 2008.

[2] S. Chen and H. shen, "Semantic-aware dummy selection for location privacy preservation," in *Proceedings of the 16th IEEE International Conference on Trust, Security and Privacy in Computing and Communications*, pp. 752–759, Aug. 2017.

[3] R. Cheng, Y. Zhang, E. Bertino, and S. Prabhakar, "Preserving user location privacy in mobile data management infrastructures," *Lecture Notes in Computer Science*, no. 4258, pp. 393–412, 2006.

[4] C. Y. Chow, M. F. Mokbel, and X. Liu, "A peer-to-peer spatial cloaking algorithm for anonymous location-based service," in *Proceedings of the 14th ACM International Symposium on Geographic Information Systems (ACM-GIS'06)*, pp. 171–178, Nov. 2006.

[5] M. Gruteser and D. Grunwald, "Anonymous usage of location-based services through spatial and temporal cloaking," in *Proceedings of the 1th International Conference on Mobile Systems, Applications, and Services*, pp. 31–42, May 2003.

[6] Y. Huang, Z. Huo, and X. F. Meng, "Coprivacy: A collaborative location privacy-preserving method without cloaking region," *Chinese Journal of Computers*, vol. 34, no. 10, pp. 1976–1985, 2011.

[7] R. H. Hwang, Y. L. Hsueh, J. J. Wu, and F. H. huang, "Social hide: A generic distributed framework for location privacy protection," *Journal of Network & Computer Applications*, no. 76, pp. 87–100, 2016.

[8] H. Kido, Y. Yanagisawa, and T. Satoh, "An anonymous communication technique using dummies for location-based services," in *Proceedings of 1st International Workshop on Security, Privacy and Trust in Pervasive and Ubiquitous Computing*, pp. 88–97, Aug. 2005.

[9] H. Kido, Y. Yanagisawa, and T. Satoh, "Protection of location privacy using dummies for location-based services," in *Proceedings of the 21st International Conference on Data Engineering Workshops*, pp. 1248, Apr. 2005.

[10] H. I. Kim, H. J. Kim, and J. W. Chang, "A secure knn query processing algorithm using homomorphic encryption on outsourced database," *Data & Knowledge Engineering*, 2017. (https://www.sciencedirect.com/science/article/pii/S0169023X17303476)

[11] J. Li, H. Y. Yan, Z. L. Liu, X. F. Chen, X. Y. Huang, and D. S. Wong, "Location-sharing systems with enhanced privacy in mobile online social networks," *IEEE Systems Journal*, vol. 11, no. 99, pp. 1–10, 2015.

[12] L. Li, J. Hua, S. Wan, H. Zhu, and F. Li, "Achieving efficient location privacy protection based on cache," *Journal on Communications*, vol. 38, no. 6, pp. 148–157, 2017.

[13] H. Liu, X. H. Li, E. M. Wang, and J. F. Ma, "Privacy enhancing method for dummy-based privacy protection with continuous location-based service queries," *Journal on Communications*, vol. 37, no. 7, pp. 140–150, 2016.

[14] H. Lu, C. S. Jensen, and L. Y. Man, "Pad: Privacy-area aware, dummy-based location privacy in mobile services," in *Proceedings of the 7th ACM International Workshop on Data Engineering for Wireless and Mobile Access*, pp. 16–27, Jan. 2008.

[15] K. Mouratidis and M. L. Yiu, "Location-sharing systems with enhanced privacy in mobile online social networks," *Proceedings of the VLDB Endowment*, vol. 5, no. 8, pp. 692–703, 2012.

[16] W. W. Ni, Z. X. Ma, and X. Chen, "Safe region for privacy-preserving continuous nearest neighbor query on road networks," *Journal of Computer Science*, vol. 39, no. 3, pp. 628–642, 2016.

[17] B. Niu, S. Gao, F. H. Li, H. Li, and Z. Q. Lu, "Protection of location privacy in continuous lbss against adversaries with background information," in *Proceedings of the 3rd International Conference on Computing, Networking and Communications (ICNC'16)*, pp. 1–6, Feb. 2016.

[18] B. Niu, Q. Li, X. Zhu, G. Cao, and H. Li, "Achieving k-anonymity in privacy-aware location-based services," in *Proceedings of the 33rd Annual IEEE International Conference on Computer Communications (INFOCOM'14)*, pp. 754–762, Apr. 2014.

[19] B. Niu, Q. H. Li, X. Y. Zhu, G. H. Cao, and H. Li, "Enhancing privacy through caching in location-based services," in *Proceedings of the IEEE Conference on Computer Communications (INFOCOM'15)*, pp. 1017–1025, Apr. 2015.

[20] Z. X. Pei, X. H. Li, H. Liu, and K. Y. Lei, "Anonymizing region construction scheme based on query range in location-based service privacy protection," *Journal on Communications*, vol. 38, no. 9, pp. 125–132, 2017.

[21] R. Shokri, G. Theodorakopoulos, P. Papadimitratos, E. Kazemi, and J.P. Hubaux, "Hiding in the mobile crowd: Location privacy through collaboration," *IEEE Transactions on Dependable and Secure Computing*, vol. 11, no. 3, pp. 269–279, 2014.

[22] Y. Sun, M. Chen, L. Hu, and Y. Qian, "Asa: Against statistical attacks for privacy-aware users in location based service," *Future Generation Computer Systems*, vol. 70, no. 2017, pp. 48–58, 2016.

[23] E. Troja and S. Bakiras, "Leveraging P2P interactions for efficient location privacy in database-driven dynamic spectrum access," *International Journal of Network Security*, vol. 17, no. 5, pp. 569–579, 2015.

[24] Y. Wu, T. Wang, and J. D. Li, "Clustering parameters selection algorithm based on density for divisional clustering process," *Control and Decision*, vol. 31, no. 1, pp. 21–29, 2016.

[25] M. B. Xie, Q. Qian and S. Ni, "Clustering based k-anonymity algorithm for privacy preservation," *International Journal of Network Security*, vol. 19, no. 6, pp. 1062–1071, 2017.

[26] P. Xie, J. Guo, and Q. Wang, "A-anonymous polygon area construction method and algorithm based on lbs privacy protection," *Journal of Information & Computational Science*, vol. 12, no. 15, pp. 5713–5724, 2015.

[27] J. Xu, X. Tang, H. Hu, and J. Du, "Privacy-conscious location-based queries in mobile environments," *IEEE Transactions on Parallel & Distributed Systems*, vol. 21, no. 3, pp. 313–326, 2010.

[28] T. Xu and Y. Cai, "Exploring historical location data for anonymity preservation in location-based services," in *Proceedings of the 27th Conference on Computer Communications*, pp. 547–555, Apr. 2008.

[29] T. Xu and Y. Cai, "Feeling-based location privacy protection for location-based services," in *Proceedings of the 2009 ACM Conference on Computer and Communications Security*, pp. 348–357, Jan. 2009.

[30] Y. Yang and R. Wang, "Rectangular region k-anonymity location privacy protection based on lbs in augmented reality," *Journal of Nanjing Normal University (Natural science)*, vol. 39, no. 4, pp. 44–49, 2016.

[31] C. Yin, R. Sun, and J. Xi, "Location privacy protection based on improved k-value method in augmented reality on mobile devices," *Mobile Information Systems*, vol. 2017, no. 12, pp. 1–7, 2015.

[32] T. H. You, W. C. Peng, and W. C. Lee, "Protecting moving trajectories with dummies," in *Proceedings of the 9th International Conference on Mobile Data Management*, pp. 278–282, June 2008.

[33] R. Yu, J. Kang, X. Huang, S. Xie, Y. Zhang, and S. Gjessing, "Mixgroup: Accumulative pseudonym exchanging for location privacy enhancement in vehicular social networks," *IEEE Transactions on Dependable & Secure Computing*, vol. 13, no. 1, pp. 93–105, 2016.

[34] H. Zhang, N. Yu, and Y. Wen, "Mobile cloud computing based privacy protection in location-based information survey applications," *Security & Communication Networks*, vol. 8, no. 6, pp. 1006–1025, 2015.

[35] X. J. Zhang, X. L. Gui, and Z. D, "Privacy preservation for location-based services: A survey," *Journal of Software*, vol. 26, no. 9, pp. 2373–2395, 2015.

[36] C. Zhou, C. Ma, and S. Yang, "Location privacy-preserving method for lbs continuous knn query in road networks," *Journal of Computer Research and Development*, vol. 52, no. 11, pp. 2628–2644, 2015.

[37] J. Zhu, B. Hu, and H. Shao, "Trajectory similarity measure based on multiple movement features," *Journal of Wuhan University (Information Science Edition)*, vol. 42, no. 12, pp. 1703–1710, 2017.

Biography

Yong-bing Zhang He is currently a Ph.D. student in Lanzhou University of Technology, and worked at school of Gansu Institute of Mechanical & Electrical Engineering. He received his master degree in electronic and communication engineering from Lanzhou University of Technology, Gansu, China, in 2015. His research interests include network and information security, privacy protection.

Qiu-yu Zhang Researcher/PhD 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.

Zong-yi Li Professor. graduated from Xi'an Jiao Tong University in 1982, and then worked at school of Gansu Institute of Mechanical & Electrical Engineering. He received his master degree in mechanical engineering from Xi'an Jiao Tong University. His research interests include advanced manufacturing technology, engineering CAD technology, expert system, Intelligent manufacturing.

Yan Yan associate professor. received her master degree in communication and information systems from Lanzhou University of Technology, Gansu, China, in 2005. She is currently a Ph.D. student in Lanzhou University of Technology. Her research interests include privacy protection, multimedia information security, uncertain information processing.

Mo-yi Zhang She is currently a Ph.D. student in Lanzhou University of Technology. She received her master degree in communication and information systems from Lanzhou University of Technology, Gansu, China, in 2010. Her research interests include Artificial intelligence, image processing and pattern recognition, robot vision.

Efficient Near-Duplicate Document Detection Using Consistent Weighted Sampling Filter

Xinpan Yuan[1], Songlin Wang[1], Cheng Peng[1], and Chengyuan Zhang[2]
(Corresponding author: Cheng Peng)
School of Computer Science, Hunan University of Technology[1]
Zhuzhou 412007, China
School of Information Science and Engineering, Central South University, Changsha, China[2]
(Email: 2561838940@qq.com)
(Received Oct. 9, 2018; Revised and Accepted Feb. 7, 2019; First Online July 16, 2019)

Abstract

Near-duplicate document detection is a problem of pursuing data pairs whose similarities are higher than the specific threshold (*e.g.*, 0.7) from the large database. Recently,Consistent Weighted Sampling algorithm (or weighted min-wise hash) and its related hashing algorithms have achieved great performances in near-duplicate detection. However, there are a large number of comparisons for data pairs, which may spend a lot of computation time and affect the performance. This paper proposes a fast consistent weighted sampling filtering algorithm to greatly reduce the calculation time by terminating the unnecessary comparisons in advance. We have proved that the filter is correct and effective through the experiment on the two synthetic data-set (UNIFORM, GAUSSIAN) and a real data (FUNDS).

Keywords: Consistent Weighted Sampling; Filtering; Near-duplicate Document Detection; Similarities

1 Introduction

Explosive information growth of Web leads to a huge amount of similar information on the Web. Research shows that 80% -90% of the data is redundant in backup and archival storage systems, and this rate is still increasing, which is a big waste [24]. These similar documents consumed a lot of storage and computation resources and reduce the efficiency of web search engines [27]. Some duplications come from plagiarism and illegal proliferation. Near-duplicate document detection [16, 22] in intellectual property protection and information retrieval has important applications. The main problem of near-duplicate detection is to find data pairs with similarity greater than the threshold from the large database.In the case that the database is very large or that the similarity computations between the pairs are very costly.

Traditionally, when comparing the similarities of two texts, most of them measure the similarity of texts into the distances from which textual feature vectors are calculated. Common text similarity measurement algorithms have cosine similarity [3], Euclidean distance [13], edit distance [18] and Jaccard coefficient [28], *etc.* These algorithms are only suitable for short texts or when the amount of data is relatively small, and cannot handle long texts and massive text data. In the face of the similarity measure of massive text data, most scholars generate K hash codes or fingerprints from K independent sample outputs, and then estimate the similarity between texts by counting the number of fingerprints equal. Such algorithms are collectively referred to as hash similarity metrics. The most representative is Minwise Hash.

Minwise Hashing [3] (or Minhash) is a Locality Sensitive Hashing, and is considered to be the most popular similarity estimation methods. Many LSH schemes have been proposed, divided between metric-driven [1, 5, 6, 14] with the goal of approximating a given distance metric, and data-driven [19, 25] where the hash functions are learned to optimize performance on a task such as classification.

Minhash keeps a sketch of the data and provides an unbiased estimate of pairwise Jaccard similarity. The algorithm is widely used for near-duplicate web page detection and clustering [9, 12] set similarity measures [2] nearest neighbor search [7] large-scale learning [11] etc. Minwise Hash can quickly and efficiently estimate the similarity of two collections. Minwise Hash requires K (commonly, K=1000) independent random permutations to deal with the datasets [26] It denotes π as a random permutation function: π: $\Omega \rightarrow \Omega$. The similarity between two non-empty sets S_1 and S_2 is defined as:

$$P_r(\min(\Pi(S_1)) = \min(\Pi(S_2)) = \frac{|S_1 \cap S_2|}{|S_1 \cup S_2|} = J(S_1, S_2).$$

It can effectively solve the problem of the time and space complexity of solving the similarity of massive data,and the generated feature fingerprints can be used for the next comparison and is widely used However, the

Minwise Hash algorithm does not consider the weight of elements in the set. When a deeper understanding of the words in the text is required, the words should be given corresponding weights. Matching of page titles should be considered more important than matching of other elements in search engines. Besides, the titles, keywords, and abstracts of a document are more important than others. In a sampling algorithm, words or phrases are selected with a low frequency of occurrence in the corpus, because common terms or phrases may represent idiomatic or spurious repetitions.

Manasse [8] introduced the concept of consistent weighted sampling (CWS), which focuses on sampling directly from some well-tailored distribution to avoid any replication. This method, unlike previous ones, could handle real weights exactly. Going a step further, Ioffe [10] proposed weight minhash scheme (WMH) which was able to compute the exact distribution of min-wise sampling leading to a scheme with worst case $O(d)$, where d is the number of non-zeros. Later, Shrivastava [20] provided an exact weighted min-wise hashing with same property as WMH but significantly faster than WMH.

Recent advances based on the idea of densification (Shrivastava & Li, 2014a; c) have shown that it is possible to compute k min-wise hashes, of a vector with d nonzeros, in mere (d; k) computations, a significant improvement over the classical $O(dk)$. These advances have led to an algorithmic improvement in the query complexity of traditional indexing algorithms based on min-wise hashing [21].

To find data pairs with similarities higher than the threshold in the big data set, the usual method of neardu-plicate detection go through the following steps:

1) Feature extraction of data in the dataset;

2) Clustering based on characteristics of the data to form data pairs worth measuring similarity;

3) Computing the similarity value of pairs using the similarity measure function (*e.g.* Minwise hash or CWS).

In this paper, our main contributions include: We design a threshold filter base on CWS and propose a faster weighted hash similarity measurement algorithm, in order to quickly and accurately calculate the similarity in large-scale datasets.

The rest of the paper is organized as follows: Section 2 discusses the related works. Section 3 describes the weighted sampling algorithm in detail. Section 4 introduces a faster similarity measurement over threshold. Experimental evaluations are presented in Section 5. Finally, Section 6 gives conclusions.

2 Feature Representation

TF-IDF is a commonly used text weighting technique for information retrieval and data mining. TF indicates the frequency with which feature item appears in document. IDF [15] represents the quantification of the distribution of feature items in the document set. TF is term frequency and IDF is inverse document frequency.

TF-IDF is a statistical method for assessing the importance of a word for a document set or one of the documents in a corpus [17]. The importance of a word increases proportionally with the number of times it appears in the file, but at the same time it decreases inversely with the frequency with which it appears in the corpus. The main idea is that if a word or phrase has a high frequency of TF in an article and rarely appears in other articles, the less the document contains the feature item and the larger the IDF, the word or phrase has a good classification ability.

For the term t_i appearing in the document d_j, its word frequency TF can be expressed as:

$$TF_{i,j} = \frac{n_{i,j}}{\sum_k n_{k,j}},$$

where $n_{i,j}$ is the number of occurrences of the word in the file, and the denominator is the sum of the occurrences of all the words in the document. The reverse document frequency IDF [23] can be expressed as:

$$IDF_i = \log \frac{|D|}{|j : i \in d_j| + 1}.$$

Where $|D|$ represents the DBLP: `conf/bigdataconf/LuoNH13` total number of files in the corpus. $|j : i \in d_j|$ indicates the number of documents containing the word t_i. So the TF-IDF of term t_i can be expressed as: TF-IDF=TF×IDF, Therefore, a document can be represented by the weighted set S={TF-IDF$_1$, TFIDF$_2$, $\cdots$, TF-IDF$_n$} after the above processing.

3 Document Clustering

The main role of document clustering is to form pairs of documents that should be matched in document set. Then, the similarity estimation is performed on the formed document pairs. The process consists of three steps [4] as shown in Figure 1.

Step 1. Calculating a sketch for every document. Generate a list of all the shingles and the documents they appear in, sorted by shingles value.The sketch is expanded into a list of ⟨shingles value, document ID⟩ pairs. The list is then sorted using the split, sort, and merge method.

Step 2. Generating a list of all the pairs of documents that share any shingles, along with the number of shingles they have in common. To do this, taking the file of sorted ⟨shingle, ID⟩ pairs and expand it into a list of ⟨ID, ID, count of common shingles⟩ triplets by taking each shingle that appears in multiple documents and generating the complete set of ⟨ID, ID, 1⟩ triplets for that shingle. Then, applying the divide, sort, merge procedure (adding the counts for

matching ID - ID pairs) to produce a single file of all triplets sorted by the first document ID.

Step3. Examining each ⟨ID, ID, count⟩ triplet and decide if the document pair exceeds our threshold for resemblance. If it does, making clusters of the document pair.

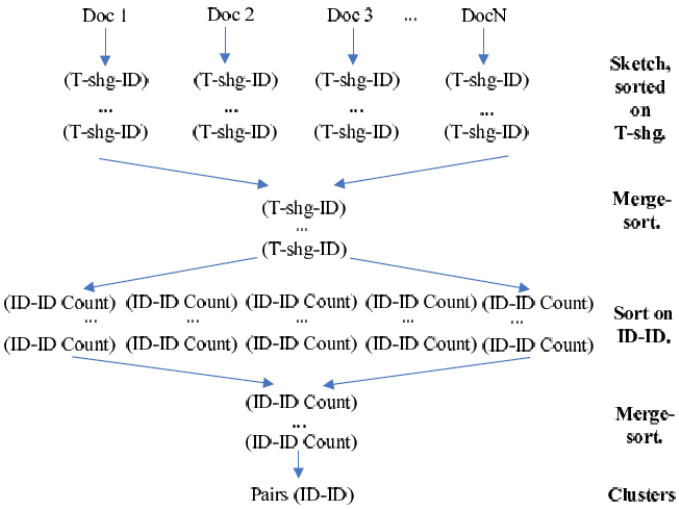


Figure 1: The cluster process

4 Weighted Sampling Algorithm

The consistent weighted sampling (CWS) [8] is a sampling scheme, sampling representatives from a weighted set such that for any non-empty weighted sets S_1 and S_2, the probability that the two choose the same sample is equal to the Jaccard similarity:

$$R = Pr[sample(S) = sample(T)] = \frac{\sum_x min(S(x), T(x))}{\sum_x max(S(x), T(x))} \quad (1)$$

Where $sample(S)$ is a representative sample value of pair (x, y) with y satisfies $0<y\leqslant S(x)$, where $S(x)$ is the weight of any element x.

When x is selected to the sample output pair, the probability of x selection is proportional to $S(x)$. and y is new generated value of $sample()$, is uniformly distributed between 0 and $S(x)$, and finally only the value of y plays a decisive similarity role, as the sample of the weightings S. As shown in Algorithm 1 is the sampling algorithm steps.

As shown in Figure 2, by sampling K from the set S, we can get $Sample(S) = \{p_1, p_2, \cdots, p_k\}$, where p_i is i-th sampled value of set S by Sequence i.

Given two weighted sets S_1 and S_2, perform K times independent samplings to generate values: $Sample(S_1)=p_{1,1}$, $\{p_{1,2}, \cdots, p_{1,k}\}$ and $Sample(S_2)=\{p_{2,1}, p_{2,2}, \cdots, p_{2,k}\}$. the CWS similarity estimator:

$$R = \frac{1}{K}\sum_{i}^{K} 1\{p_{1,i} == p_{2,j}\}. \quad (2)$$

Algorithm 1 Sampling algorithm

1: Input: Given a non-empty weighted set S, $S(x)$ is the weight of any element x
2: Output: $pair(x, y)$
3: **Step1:** GenerateSampleSequences$(x, k, salt)$
4: (1) Generate a number of points on the interval(0, ∞), each point is $2k, k \in (0, \infty)$;
5: (2) Take a k such that $x \in (2^{k-1}, 2^k]$;
6: (3) Generate a random number random within the range of [0, 1] and calculate sample $= 2^k \divideontimes random$;
7: **while** sample $> 2^{k-1}$: **do**
8: save $sample$;
9: $sample = sample \divideontimes random$;
10: **end while**
11: **Step2:** Active indices: (y, z) = **ActiveIndices** $(x, S(x), salt)$
12: According to the sequences obtained in Step 1, determine the greatest value $y \leq S(x)$ and the least value $z > S(x)$ in expected constant time. That is, z=the first right value of the sequence greater than $S(x)$; y=the first left value of the sequence less than $S(x)$.
13: **Step3:** Uniform sampling
14: (1) Define $h_{max} = 0$, $x_{max} = null$, $y_{max} = 0$;
15: (2) for each x in $S(x)$:
16: a. Random variable β is within the range of [0, 1];
17: b. $cdf_z(a) = a^z + a^z z \ln(1/a) = \beta$;
18: c. $h = cdf^{-1}(\beta)$.
19: (3) Take the value x and y with the maximum h_{max} of all h, and return $(x_{max}y_{max})$.
20: After the above three steps, the final pair value of (x_{max}, y_{max}) is the first sampled value of set S by $Sequence_1$.
21: End

5 Faster Similarity Measurement Over Threshold

Most practical applications scenes such as near-duplicate detection, clustering, and nearest neighbor search, only care about the similarity is greater than a certain high threshold [30] $T0$ (*e.g.* $T0$=0.8) between pairwise data, as shown in Figure 3. In this application scenario, in order to quickly and accurately obtain the similar data over the specific threshold, we design a filter of CWS to terminate the unnecessary matching process in advance.

5.1 Observation Threshold

At first, the filter divides the entire comparison process of for each $p_{1,i} = p_{2,i}$ in Equation (2) into several subprocesses with observation points $\{k_1, k_2, \cdots, k_j, \cdots, k_n\}$, $(0 < k_i \leqslant K, 0 < i \leqslant n)$. The similarity of Equation (2)

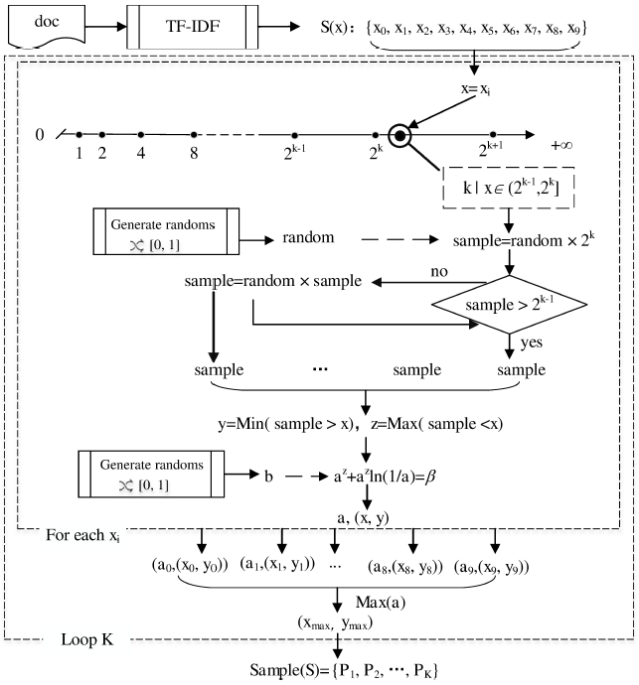


Figure 2: The sampling process

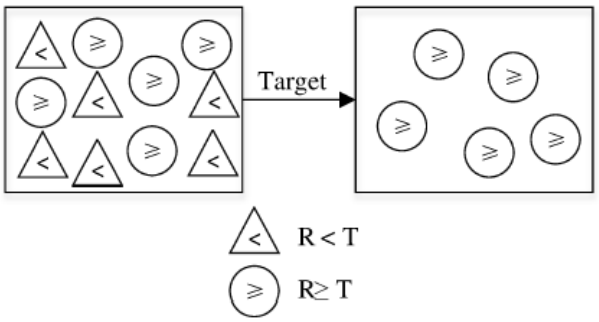


Figure 3: A filter with similarity $R \geq T$

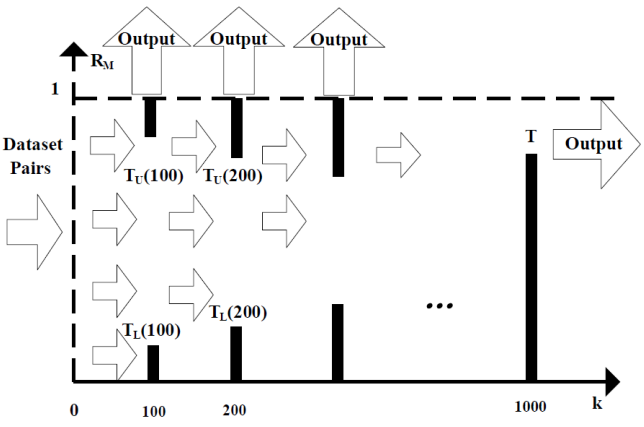


Figure 4: Schematic diagram of filtering process

at the j-th observation point is different from the similarity with argument K, because the more sampling times, the accuracy of similarity is higher. Therefore, in the case of a small k_j, the similarity of the calculations is meaningless. However, We can set the threshold to be higher than the threshold T when the number of samples is K, and set another threshold to filter out the data pairs that do not need to be compared in advance.

As shown in Figure 4, the filter defines the lower bound threshold $T_L(k)$ and the upper bound threshold $T_U(k)$ for any k observation, then the filter can output the pairwise data that exceeds the real similarity over T, when the point of observation at k is higher than $T_U(k)$. And in advance filtering the pairwise data, when the point of observation at k is less than $T_L(k)$, thereby speeding up the comparison process. The $T_U(k)$ and $T_L(k)$ can be found by hypothesis testing and small probability events.

5.2 How to Setting Threshold At K Observation

The total number of comparisons K is reduced to k, and the random variable X at the observation point k is equal to the number of equal terms of $Sample(S_1)$=$\{p_{1,1}, p_{1,2}, \cdots, p_{1,k}\}$ and $Sample(S_2)$=$\{p_{2,1}, p_{2,2}, \cdots, p_{2,k}\}$. The definition of X is shown in Equation (3).

$$X = \sum_{i}^{k} 1\{p_{1,i} = p_{2,j}\} \tag{3}$$

Obviously, X obeys the binomial distribution $X \sim B(n, R)$; the probability function $\Pr(X=m)$ of random variable X is:

$$Pr(X = m) = \binom{k}{i} R^m (1-R)^{K-m} \tag{4}$$

The probability $\Pr(X \leqslant m)$ is: $Pr(X \leq m) = \binom{k}{1} R^1 (1-R)^{k-1} + \binom{k}{2} R^2 (1-R)^{k-2} + \cdots + \binom{k}{i} R^i (1-R)^{k-i} + \cdots + \binom{k}{m} R^m (1-R)^{k-m}$.

Then $\Pr(X \leqslant m)$ and $\Pr(X > m)$ is:

$$Pr(X \leqslant m) = \sum_{i=0}^{m} \binom{k}{i} R^i (1-R)^{k-i}$$

$$Pr(X > m) = \sum_{i=m+1}^{k} \binom{k}{i} R^i (1-R)^{k-i}$$

First make assumptions and use appropriate statistical methods to determine the probability of hypothesis. If the probability is high, although the assumption is not necessarily correct, if the possibility is small, the assumption is absolutely wrong.

The threshold is expressed as T and small probability is expressed as e. The method of test is as following:

1) Hypothesis.

Hypothesis $\mathcal{H}1$: The similarity R is greater than threshold T.

Hypothesis $\mathcal{H}2$: The similarity R is equal or less than threshold T.

The obvious is the $\mathcal{H}\mathbf{1} = \neg\mathcal{H}\mathbf{2}$, and $\neg\mathcal{H}\mathbf{1} = \mathcal{H}\mathbf{2}$.

2) Test hypothesis $\mathcal{H}1$: Test if the probability $\Pr(X \leqslant m)$ of variable X is small enough to be called a small probability event e? If

$$\sum_{i=0}^{m} \binom{k}{i} T^i (1-T)^{k-i} = e$$

Then the hypothesis $\mathcal{H}\mathbf{1}$ is wrong, that is $\neg\mathcal{H}\mathbf{1}$ is correct, and $\mathcal{H}\mathbf{2}$ is correct.

The same reason can be tested hypothesis $\mathcal{H}\mathbf{2}$. If

$$\sum_{i=m+1}^{k} \binom{k}{i} T^i (1-T)^{k-i} = e$$

Then the hypothesis $\mathcal{H}\mathbf{2}$ is wrong, that is $\neg\mathcal{H}\mathbf{2}$ is correct,and $\mathcal{H}\mathbf{1}$ is correct.

But how to setting threshold at k observation?

Let the total comparison number K=1000 (unrelated parameter), the observation comparison number k=100, the small probability e=1E-5, the threshold T=0.8. there are $\Pr(X \leqslant m)$ and $\Pr(X > m)$ as shown in Table 1 and Figure 5.

Table 1: Probability distribution of $\{X \leq m\}$ and $\{X > m\}$when $T = 0.8, k = 100$

m	$P_r\{X \leqslant m\}$	$P_r\{X > m\}$	m	$P_r\{X \leqslant m\}$	$P_r\{X > m\}$
5	$1.29E-61$	1	55	$4.22E-09$	1
10	$6.48E-53$	1	60	$1.29E-06$	0.999999
15	$1.57E-45$	1	65	0.000147	0.999853
20	$4.89E-39$	1	70	0.006059	0.993941
25	$3.09E-33$	1	75	0.087475	0.912525
30	$5.04E-28$	1	80	0.440538	0.559462
35	$2.49E-23$	1	85	0.871494	0.128506
40	$4.09E-19$	1	90	0.994304	0.005696
45	$2.4E-15$	1	95	0.999981	$1.87E-05$
50	$5.18E-12$	1	100	1	$2.04E-10$

As shown in Table 1, we can observe that m=60 and m=95 are located between a small probability and not a small probability, so we can define m=95 as m_{uper} and m=60 as m_{lower}. According to the small probability e=1E-5, if there is R>95% at the observation point of k=100, then we can completely determine R>80%. The same reason, if there is R<60% at the observation point k, then we can completely determine R<80%, so so we can filter out the unnecessary comparison process in advance.

$$\begin{aligned} T_U(k) &= m_{uper}/k \\ T_L(k) &= m_{lower}/k \end{aligned}$$

We try to give a formal description from the perspective of conditional probability. Let event $A = \{R_K > T\}$, $B = \{R_k > T_U(k)\} = \{X > m_{uper}\}$, our goal is $Pr(A|B) = \frac{Pr(A|B)}{Pr(B)=1}$. If $\Pr(B)\to 0 \Rightarrow \Pr(A|B)=1$. The same reason, let event $C=\{R_K<T\}$, $D=\{R_k<T_L(k)\}=\{X<m_{lower}\}$. If $\Pr(D)\to 0 \Rightarrow \Pr(C|D)=1$.

With threshold T and observation points$\{k_1, k_2, \cdots, k_j, \cdots, k_{n-1}, k_n\}$, $(0 < k_i \leqslant K, 0 < i \leqslant n)$, there is lots of $T_U(k)$ and $T_L(k)$ to be settled into the filter, as shown as Figure 6.

The calculation process of the consistent weighted sampling filtering algorithm proposed in the paper is shown in Algorithm 2.

Algorithm 2 The calculation process of the weighted sampling filtering algorithm

Input: Weighted set $\{(S_1, S_2), (S_3, S_4), \cdots, (S_{2n-1}, S_{2n})\}$
Output: The similarity R of the set pairs is greater than the threshold T $\{(S_{2i-1}, S_{2i}), (S_{2i-1}, S_{2i}) | R(S_{2i-1}, S_{2i}), (1 \leqslant i \leqslant n)\}$
(1) Setting parameters Similarity threshold T, observation point $\{k_1, k_2, \cdots, k_i, \cdots, k_{n-1}, k_n\}$, Number of samples K, Small probability e;
(2) Using the CWS algorithm to generate corresponding fingerprints, for example Sample$(S_1) = \{p_{1,1}, p_{1,2}, \cdots, p_{1,K}\}$ and Sample$(S_2) = \{p_{2,1}, p_{2,2}, \cdots, p_{2,K}\}$;
(3) At the observation point $k = k_i$, Calculate the upper and lower thresholds $T_U(k) = m_{uper}/k, T_L(k) = m_{lower}/k$;
(4) For each fingerprint pair, compare the first k fingerprints, calculate the similarity $R(k)$ at the observation point;
if $R(k) \leqslant T_L(k)$ **then**
    The fingerprint pair is excluded in advance, and the corresponding weighted set pair;
else if $R(k) \geqslant T_U(k)$ **then**
    Output fingerprint pairs in advance, and corresponding weighted set pairs;
else
    $i++; k = k_i$; Go to (3) until $k_i = K$
end if

6 Experimental Evaluation

6.1 Experimental Dataset

We select the data pair set from the FUNDS set as the source data set. **FUNDS:** Text application for the Natural Science Foundation project, approximately 100,000 documents. Considering that real-world data sets are usually distributed between Gaussian and uniform distribution, in order to make the experimental results more reliable, we synthesize datasets of uniform distribution and Gaussian distribution from the FUNDS dataset. Finally we synthesize Gaussian, uniform and FUNDS datasets,

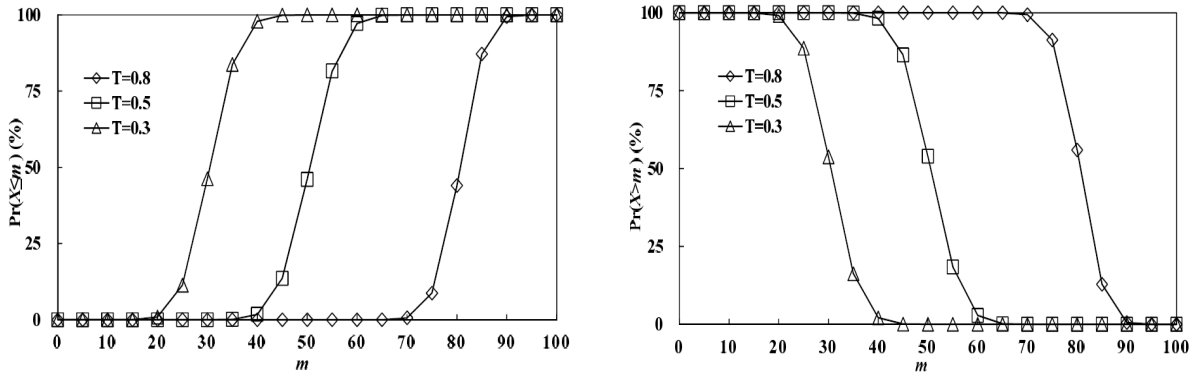


Figure 5: (a) Pr($X>m$) and (b) Pr($X\leqslant m$) varies on m

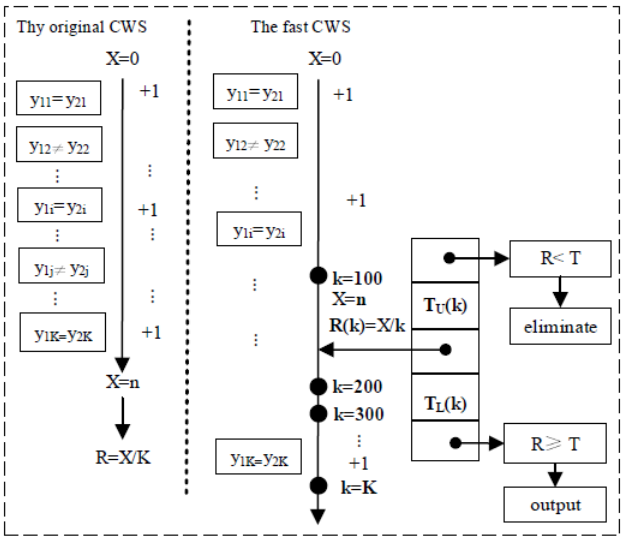


Figure 6: The fast weighted hash similarity measure filter

each with 1000 data pairs. Finally we synthesize three data sets, each with 1000 data pairs, respectively:

1) **UNIFORM** distribution dataset;
2) **GAUSSIAN** distribution dataset;
3) **FUNDS** dataset. The similarity distribution of the data set is shown in Figure 7.

6.2 Precision and Recall of Filter

According to the small probability theory, the filter does not change the estimation accuracy of the CWS algorithm. By Looking at the possibility of testing correctly filtering through actual data, the accuracy and recall of $T_U(k)$ filter is:

$$Accuracy(T_U(k)) = \frac{|R_k > T_U(k)| \bigcap |R_k > T|}{|R_k > T_U(k)|}$$

$$Recall(T_U(k)) = \frac{|R_k > T_U(k)| \bigcap |R_k > T|}{|R_k > T|}$$

Where T is the similarity threshold, K is the number of samples, R_k is the similarity at the observation point k (*e.g.* k=100), R_K is the similarity estimate of CWS (*e.g.* K=1000). And the $T_L(k)$ filtering accuracy and recall rate:

$$Accuracy(T_L(k)) = \frac{|R_k < T_L(k)| \bigcap |R_k < T|}{|R_k < T_L(k)|}$$

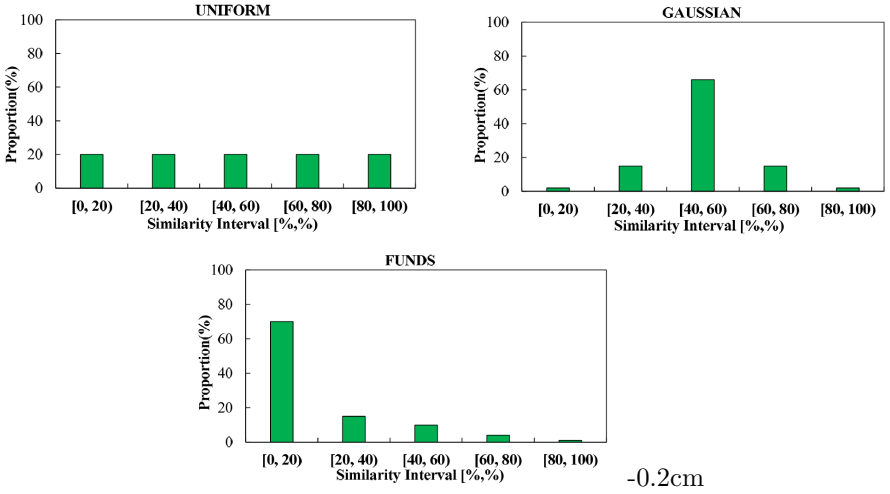
$$Recall(T_L(k)) = \frac{|R_k < T_L(k)| \bigcap |R_k < T|}{|R_k < T|}$$

As shown in the Figure 8, let the observation point k=100, the threshold T=0.8, 0.5, and 0.3, and we have following discussion.

1) Both $T_U(k)$ and $T_L(k)$ have a dividing point of accuracy, for example, when T=0.8 the accuracy of $T_U(k)$ is low in the interval [0,90], in [90,100] will certainly filter success, accuracy is 100%. The dividing point is also the dividing point between non-small probability and small probability.

2) For UNIFORM and GAUSSIAN dataset, when the accuracy is 100%, the recall is not high. It means that filtering accuracy is high, but there are still omissions, so we need to continue setting thresholds at subsequent observation points (*e.g.* k=200, 300 and so on). However, it is easy to observe that certain $T_L(k)$ values in the actual data set (FUNDS) can achieve high accuracy and recall rates because there are a large number of low similarity data in the FUNDS.In this way, a large number of data can be filtered out at the first threshold, thus reducing the time of comparison.

3) So how do we set the threshold at the observation point? As shown in the Figure 8, the setting of the observation point threshold is not related to the distribution of the data set, and determined by the principle of small probability. It can be clearly seen from any picture that the thresholds are the same. The dividing point is the best position to set the threshold.



-0.2cm

Figure 7: The similarity distribution of datasets

On the one hand, it guarantees 100% accuracy and on the other hand increases the recall rate as much as possible.

6.3 Filter Rate

As show in Figure 9, for different data distributions, there is no threshold to ensure 100% accuracy and recall rate, however, the remaining data can be filtered at subsequent observation points (for example, k = 200, 400, 600, *etc.*), and the overall recall rate will be 100%. let's look at the amount of filtering, described in terms of the proportion of the filtered data to the total data. And the filter rates of $T_U(k)$ and $T_L(k)$ are:

$$\begin{aligned} FilteringRate(T_U(k)) &= \frac{|R_k > T_U(k)|}{N} \\ FilteringRate(T_L(k)) &= \frac{|R_k < T_L(k)|}{N} \end{aligned}$$

Where N is the total number of pairs.

Let the observation point k=100,200,400,600,800, the threshold T=0.8, 0.5, and 0.3. As shown in the Figure 9, Different data sets have different amounts of filtering under the influence of $T_U(k)$ or $T_L(k)$. For uniform and Gaussian data sets, $T_L(k)$ plays a major role in filtering when the threshold is high ($T = 0.8$). When the threshold is low (T=0.3), $T_U(k)$ plays a major filtering role. But for datasets (such as FUNDS) that are mostly low similarity data, when the threshold is $T = 0.8$, $T_L(k)$ can filter out more than 90% of the data at the observation point k = 100. Even at the threshold of $T = 0.3$, it can filter out about 50% of the data.

In most practical applications, only high similar data is concerned, that is, a large threshold is set, but there is a large amount of low similarity data in the data set, so that the filter can play a greater role.

6.4 Time Cost

Based on the above analysis of the filtration rate, we can expect that the calculation time will be greatly reduced. CWS has to complete the comparison of 100 million data pairs (each CWS sampling K=1000), so it consumes almost the 1011 ($100\times106\times K$) comparisons in total.

As shown in the Figure 10, abscissa describes different distributed data sets, and the ordinate is CPU time. We still choose three typical thresholds of T=0.8, 0.5, and 0.3 to measure time. The time consumed by different thresholds for the same data set is different, depending on the filtering rate. At the same time, different dataset distributions will produce different filtering rates, for example, it can be clearly observed that the FUNDS data set has a high filtering rate, so its calculation consumes the least amount of time. **Figure 10: CPU time cost of three distribution dataset**

Experiments show that the algorithm can guarantee 100% accuracy and increase the recall rate as much as possible. The remaining filtered data can be used at subsequent observation points (for example, k = 200, 400, 600, *etc.*), and the overall recall rate is 100%. For a large number of low similarity real data, accompanied by high threshold queries, the filter reduces the 85% of comparison, compared with the original CWS algorithm.

6.5 Application

Known from the beginning of January 2018, the document's number of the National Natural Resources Fund of China is about 1.2 million. The pairs number of clustering was about 100 billion to check the similarity. According to the consistent weight sampling algorithm (do not use the optimization algorithm proposed in this paper), it takes about 5000s to complete the near-duplication detection of 100 billion pairs. However, the time taken by the CWS Filter algorithm proposed in this paper is reduced to about 280s. This will greatly reduce the time

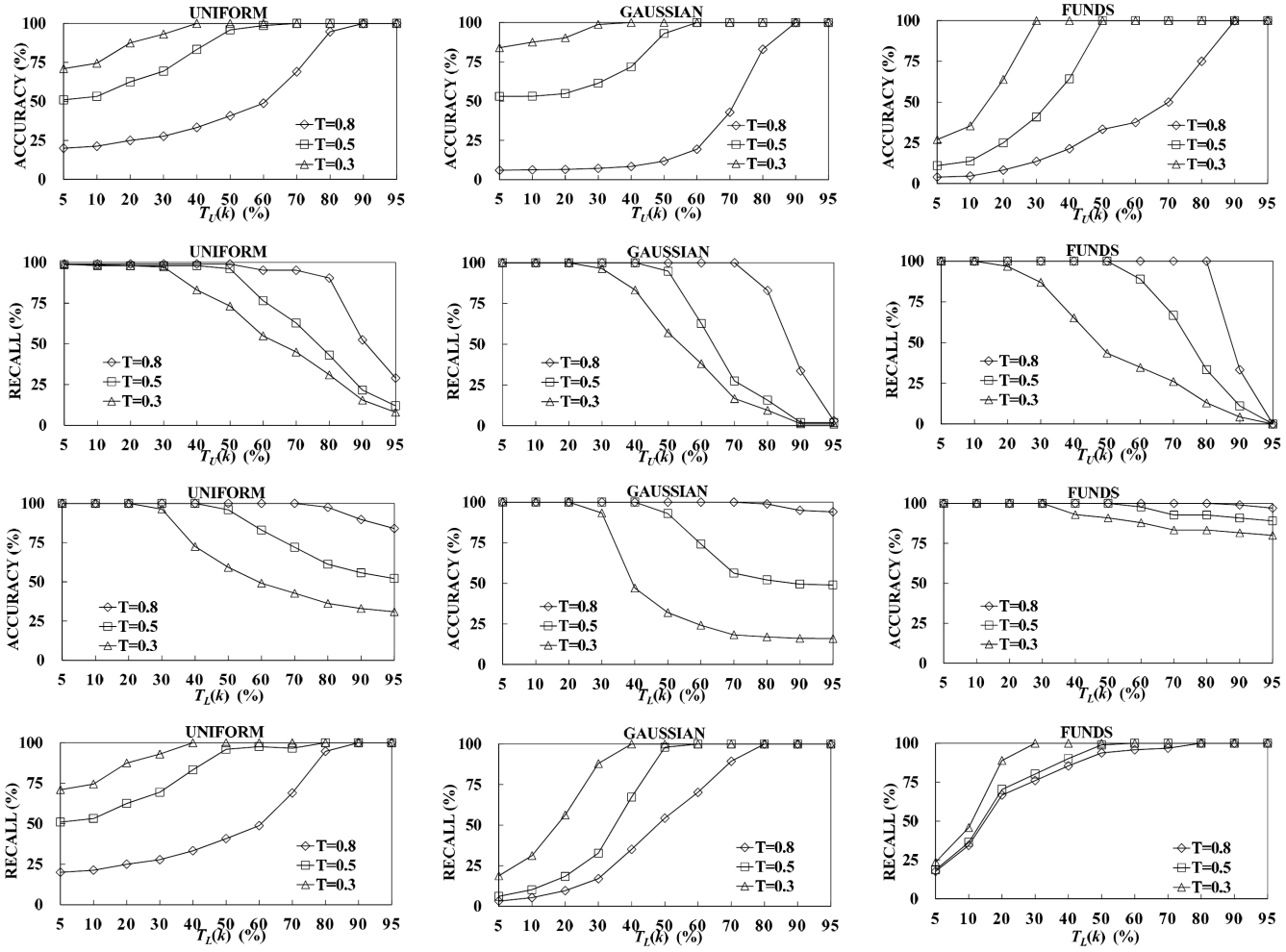


Figure 8: Accuracy and recall of three distribution dataset

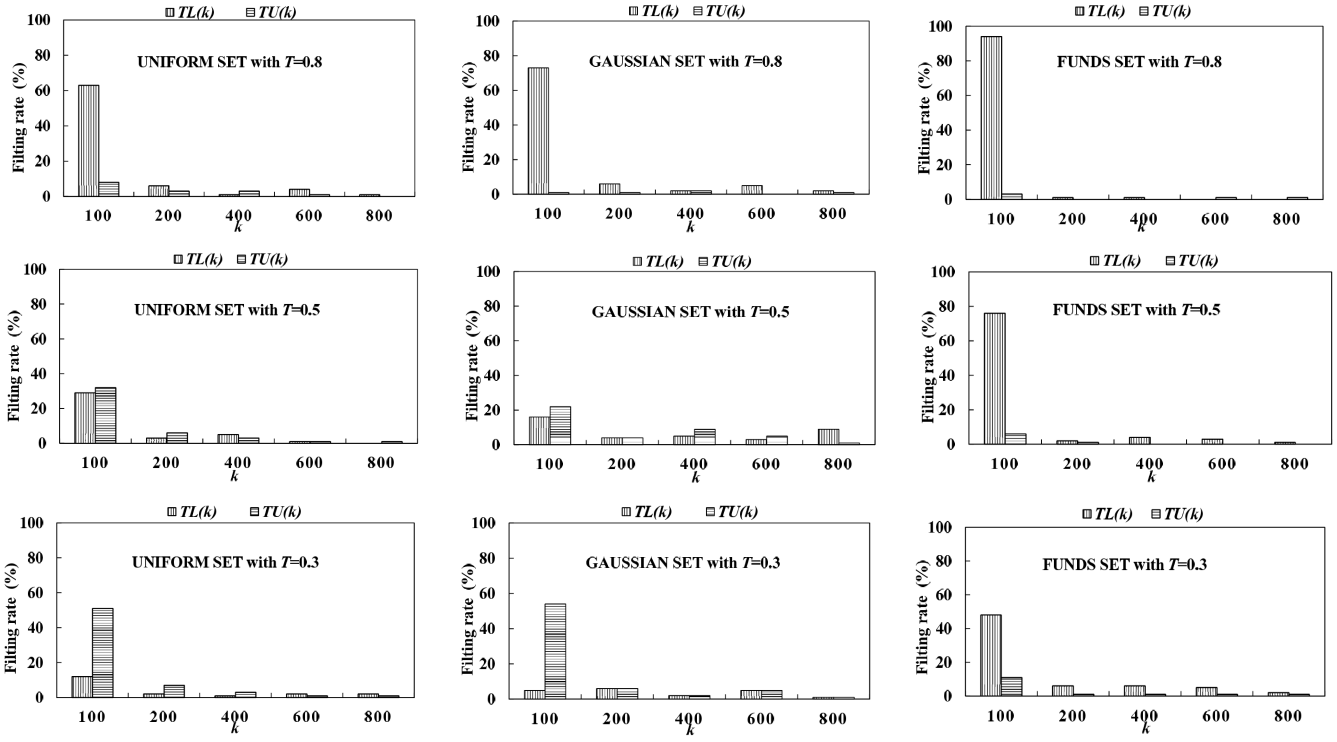


Figure 9: Filtering rate of three distribution dataset

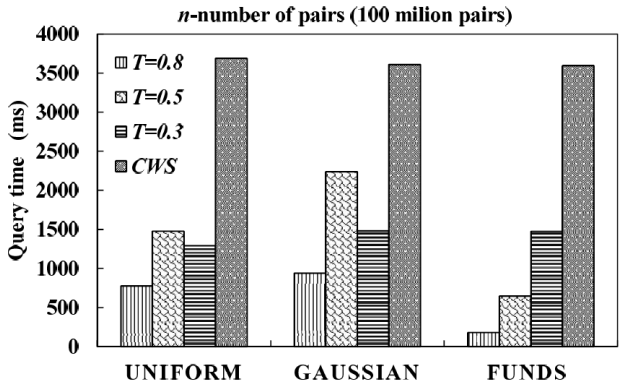


Figure 10: CPU time cost of three distribution dataset

consumption compared to the previous calculations.

7 Conclusions

In this paper, we combine binomial distribution with small probability event and propose a fast consistent weighted hash similarity measurement over threshold. It greatly reduces the calculation time by terminating the unnecessary comparison in advance. Our experimental results are based on the two synthetic dataset (UNIFORM, GAUSSIAN) and a real data(FUNDS), which proves that the filter is effective and correct.

Acknowledgments

This research was funded by National Natural Science Foundation of China [61402165, 61871432,61702560], the Key R & D programs of Hunan Province (No.2016JC2018) and Natural Science Foundation of Hunan Province [2018JJ2099,2018JJ3691].

References

[1] A. Andoni and P. Indyk, "Near-optimal hashing algorithms for approximate nearest neighbor in high dimensions," *Communications of the ACM*, vol. 51, no. 1, pp. 117–122, 2008.

[2] R. J. Bayardo, Y. M. Ma, and R. Srikant, "Scaling up all pairs similarity search," in *Proceedings of the 16th International Conference on World Wide Web (WWW'07)*, pp. 131–140, 2007.

[3] T. Bohman, C. Cooper, and A. M. Frieze, "Min-wise independent linear permutations," *The Electronic Journal of Combinatorics*, vol. 7, 2000. (`https://www.combinatorics.org/ojs/index.php/eljc/article/view/v7i1r26`)

[4] D. Brodic, A. Amelio, and Z. N. Milivojevic, "Clustering documents in evolving languages by image texture analysis," *Applied Intelligence*, vol. 46, no. 4, pp. 916–933, 2017.

[5] M. Charikar, "Similarity estimation techniques from rounding algorithms," in *Proceedings on 34th Annual ACM Symposium on Theory of Computing*, pp. 380–388, 2002.

[6] M. Datar, N. Immorlica, P. Indyk, and V. S. Mirrokni, "Locality-sensitive hashing scheme based on p-stable distributions," in *Proceedings of the 20th ACM Symposium on Computational Geometry*, pp. 253–262, 2004.

[7] X. K. Feng, J. T. Cui, Y. F. Liu, and H. Li, "Effective optimizations of cluster-based nearest neighbor search in high-dimensional space," *Multimedia Systems*, vol. 23, no. 1, pp. 139–153, 2017.

[8] B. Haeupler, M. S. Manasse, and K. Talwar, "Consistent weighted sampling made fast, small, and easy," *CoRR*, vol. abs/1410.4266, 2014.

[9] M. R. Henzinger, "Finding near-duplicate web pages: A large-scale evaluation of algorithms," in *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 284–291, 2006.

[10] S. Ioffe, "Improved consistent sampling, weighted minhash and L1 sketching," in *The 10th IEEE International Conference on Data Mining*, pp. 246–255, 2010.

[11] B. Joshi, F. Iutzeler, and M. R. Amini, "Large-scale asynchronous distributed learning based on parameter exchanges," *International Journal Data Science and Analytics*, vol. 5, no. 4, pp. 223–232, 2018.

[12] J. P. Kumar and P. Govindarajulu, "Near-duplicate web page detection: An efficient approach using clustering, sentence feature and fingerprinting," *International Journal of Computational Intelligence Systems*, vol. 6, no. 1, pp. 1–13, 2013.

[13] L. H. Lee, C. H. Wan, R. Rajkumar, and D. Isa, "An enhanced support vector machine classification framework by using euclidean distance function for text document categorization," *Applied Intelligence*, vol. 37, no. 1, pp. 80–99, 2012.

[14] P. Li, A. Christian, K. Ning, and W. H. Gui, "B-bit minwise hashing for estimating three-way similarities," in *Advances in Neural Information Processing Systems 23*, pp. 1387–1395, 2010.

[15] J. Long, Q. F. Liu, X. P. Yuan, C. Y. Zhang, and J. F. Liu, "A filter of minhash for image similarity measures," *JACIII*, vol. 22, no. 5, pp. 689–698, 2018.

[16] X. Luo, W. A. Najjar, and V. Hristidis, "Efficient near-duplicate document detection using fpgas," in *Proceedings of the IEEE International Conference on Big Data*, pp. 54–61, 2013.

[17] P. D. Qin, W. R. Xu, and J. Guo, "A novel negative sampling based on TFIDF for learning word representation," *Neurocomputing*, vol. 177, pp. 257–265, 2016.

[18] D. C. Reis, P. B. Golgher, A. S. Silva, and H. F. Laender, "Automatic web news extraction using tree edit distance," in *Proceedings of the 13th International Conference on World Wide Web (WWW'04)*, pp. 502–511, 2004.

[19] R. Salakhutdinov and G. E. Hinton, "Semantic hashing," *International Journal of Approximate Reasoning*, vol. 50, no. 7, pp. 969–978, 2009.
[20] A. Shrivastava, "Exact weighted minwise hashing in constant time," *CoRR*, vol. abs/1602.08393, 2016.
[21] A. Shrivastava, "Optimal densification for fast and accurate minwise hashing," in *Proceedings of the 34th International Conference on Machine Learning (ICML'17)*, pp. 3154–3163, 2017.
[22] J. H. Wang and H. C. Chang, "Exploiting sentence-level features for near-duplicate document detection," in *Proceedings of 5th Asia Information Retrieval Symposium on Information Retrieval Technology, (AIRS'09)*, pp. 205–217, 2009.
[23] L. Wu, Y. Wang, J. B. Gao, and X. Li, "Deep adaptive feature embedding with local sample distributions for person re-identification," *Pattern Recognition*, vol. 73, pp. 275–288, 2018.
[24] Z. Y. Wang, Y. Lu, and G. Z. Sun, "A policy-based de-duplication mechanism for securing cloud storage," *International Journal of Electronics and Information Engineering*, vol. 2, no. 2, pp. 70–79, 2015.
[25] Y. Weiss, A. Torralba, and R. Fergus, "Spectral hashing," in *Advances in Neural Information Processing Systems 21*, pp. 1753–1760, 2008.
[26] X. P. Yuan, J. Long, Z. P. Zhang, Y. Y. Luo, H. Zhang, and W. H. Gui, "F-fractional bit minwise hashing," *Journal of Software*, vol. 7, no. 1, pp. 228–236, 2012.
[27] C. Y. Zhang, Y. Zhang, W. J. Zhang, and X. I. Lin, "Inverted linear quadtree: efficient top K spatial keyword search," *IEEE Transactions on Knowledge and Data Engineering*, vol. 28, no. 7, pp. 1706–1721, 2016.
[28] K. Zheng, H. Su, B. L. Zheng, S. Shang, J. J. Xu, J. J. Liu, and X. F. Zhou, "Interactive top-k spatial keyword queries," in *IEEE of 31st International Conference on Data Engineering (ICDE'15)*, pp. 423–434, 2015.

Biography

Xinpan Yuan. He was born in Hunan Province, China. He received Ph.D. degree in computer science from the Central South University, in 2012. Currently, he is a lecturer in School of Computer Science of Hunan University of Technology, China. His research interests include Information Retrieval, Data Mining and NLP.

Songlin Wang. He was born in Henan Province, China. He received B.S. degree in computer science from Hunan University of Technology, in 2017. Currently, he is a master student in School of Computer Science of Hunan University of Technology, China. His current research interests include Information Retrieval and NLP.

Cheng Peng. He was born in Hunan Province, China. He received Ph.D. degree in computer science from the Central South University, in 2013. Currently, he is a lecturer in School of Computer Science of Hunan University of Technology, China. His research interests include software and Big Data.

Chengyuan Zhang. He was born in Hunan Province, China. He received PhD degree in computer science from the University of New South Wales. Currently, he is a lecturer in School of Information Science and Engineering of Central South University, China. His main research interests include information retrieval, query processing on spatial data and multimedia data.

A Provably Secure Group Authentication Protocol for Various LTE Networks

Boriphat Kijjabuncha[1] and Pipat Hiranvanichakorn[2]
(Corresponding author: Boriphat Kijjabuncha)
Graduate school of Applied Statistics, National Institute of Development Administration[1]
118 Moo 3, Serithai Road, Klong-Chan, Bangkapi, Bangkok 10240, Thailand
Suksoomboon Laboratory[2]
242 Tanalai Road, Amphoe Muang Chiang Rai, Chiang Rai 57000, Thailand
(Email: boriphat.k@gmail.com)
(Received Mar. 22, 2018; revised and accepted Sep. 2, 2018)

Abstract

Group authentication is beneficial for group work in the Long Term Evolution (LTE) networks because it reduces the traffic of networks. For practical use, members of a group should be able to come from different network providers. In addition, while some group members use a network service, others may use other network services. Although the group members are in different networks, they should be able to work together. To fulfill these needs, we propose a secure group authentication protocol (SE-GA) in which each group member uses his/her long term private key and public key to create shared secret (keys) with network devices, such as Home and mobile management entity (MME). These shared keys are computed by using the Diffie-Hellman key exchange and are utilized in the authentication process. By using this technique instead of pre-shared keys between mobile devices and network devices, SE-GA is flexible and scalable. In SE-GA, only the first member in a MME's area has to authenticate himself/herself with the Home, while the remaining members in the area can authenticate directly with the MME. This reduces the network traffic. In this paper, authentication proof is also given using the well-known BAN logic, and the security of the protocol is analyzed and compared with some protocols.

Keywords: BAN Authentication Logic; Diffie-Hellman Key Exchange; Group Authentication; LTE Network

1 Introduction

The research group model helps users to work together with their group even though they live in different LTE networks. However, group communication needs security management to control any risks occurred in the system and protect against unauthorized users causing a system failure. Thus, network applications need privacy, confidentiality, integrity, authentication methods to protect their information from unauthorized access.

In the mobile environment, in order to use services of a network, mobile equipment (smart phones, smart watches, laptops, *etc.*) have to authenticate themselves with their home networks (HNs). However, if several mobile equipment in the same group authenticate with their HNs at the same time the traffic of the network will be crowded. This can reduce the stability of the system, and the performance of the network decreases. Therefore, an efficient group authentication protocol is needed in the group model.

Recently, several research works have been studied on group communication and authentication [1, 6, 7, 10–15, 17, 19]. In 2009, Ou *et al.* [16] proposed a Cocktail protocol with authentication and key agreement (Cocktail-AKA) on the Universal Mobile Telecommunications System (UMTS). The protocol allows a service network (SN) to calculate the medicated authentication vectors (MAV) in advance. MAV is calculated only once and can be reused. The MAV is used with prescription authentication vector (PAV) to produce many effective authentication vectors (AVs) for mutual authentication with the mobile stations (MSs). PAV is calculated from home environment (HE). Even though the protocol can reduce computational overhead on the HE and communication overhead for delivering the AVs, the protocol has some weakness which cannot resist denial-of-service attack (DoS attack) as described by Wu *et al.* [20]. In 2012, Cao *et al.* [3] proposed a group-based authentication scheme and key agreement for Machine Type Communication (MTC) in LTE network. In the protocol, the traffic of authentication is crowded and the cost of cryptographic computing is high because MTC devices may be simultaneously authenticated by the network. Then this protocol may not be suitable for mobile devices as discussed by Lai *et al.* [8]. In the same year, Chen *et al.* [4] proposed a group-based authentication and key agreement (G-AKA) protocol for

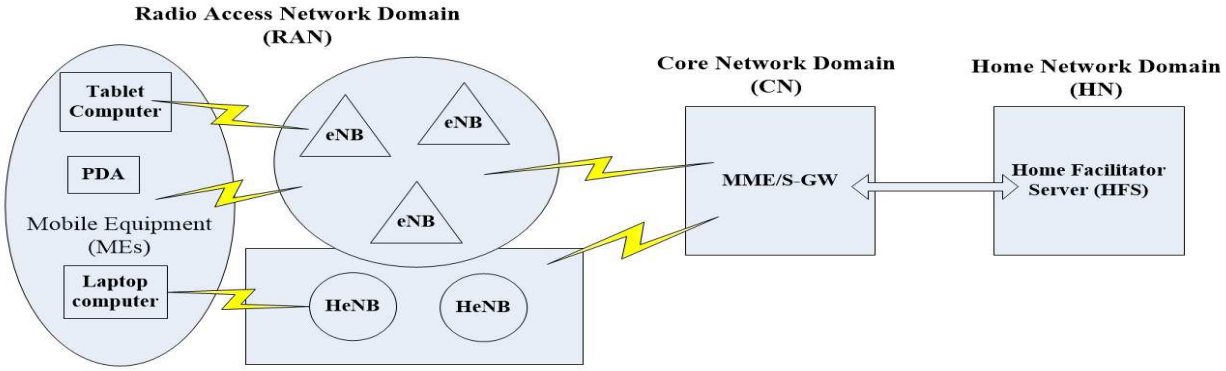


Figure 1: LTE network architecture

mobile stations (MSs) roaming from the same home network to a serving network. However, the protocol has some vulnerability such as man-in-the-middle attack as discussed by Lai *et al.* [9]. In 2013, Lai *et al.* [8] have introduced a secure and efficient group authentication and key agreement protocol (SE-AKA) which was supposed to be more secure than the evolved packet system authentication and key agreement (EPS-AKA) protocol proposed in the LTE project. In the protocol, the first mobile equipment (ME) uses its secret key to authenticate itself with its Home. Each remaining ME uses a group key and a synchronization value (SV) to authenticate itself with the service MME. However, this protocol has some weakness because a group member can be disguised by other members in the group as discussed in Section 3. In 2016, Lai *et al.* [9] proposed the group-based lightweight authentication scheme for resource-constrained machine to machine communication (GLARM). The protocol can reduce the MME overhead because the group leader collects all authentication messages from the group's members and communicates with the MME. However, as the protocol needs a group leader to send and response messages with the MME, if the group leader has some problems then the authentication process fails. Furthermore, the scope of this work is limited that all members of the group need to be in the same service network. In real work, there may be some situation that some members of the group are in different service networks.

In this paper, we propose a secure group authentication protocol (SE-GA) which makes use of users' long-term public and private keys to create secret keys with network nodes such as Home and MME. The shared keys are computed by using the Diffie-Hellman key exchange protocol based on ECC. By this way, the authentication process is flexible and scalable, and it makes group authentication easy even though group members are on different networks. In the protocol, only the first member in an MME's area has to authenticate himself/herself with the Home, while the remaining members in the area can authenticate directly with the MME. Thus SE-GA protocol can reduce network traffic. In addition, we introduce a proof for group authentication by using the well-known BAN authentication logic [2]. We have also analyzed the security of SE-GA and compared the features of the protocol with other works. From the analysis, we found SE-GA outperforms many of the past.

The rest of this paper is organized as follows. Section 2 provides some preliminaries for LTE network and elliptic curve cryptography. In Section 3, we discuss the security analysis of some previous work. SE-GA protocol is described in Section 4, and authentication proof by using BAN logic is shown in Section 5. Section 6 provides the security analysis of the protocol against some well-known attacks. The conclusion is drawn in the last section, Section 7.

2 Preliminaries

2.1 LTE Network

The LTE network architecture can be classified into 3 domains, including radio access network (RAN) domain, core network (CN) domain, and home network (HN) domain, respectively. As demonstrated in Figure 1, the network includes entities as shown in Table 1. The network is described according to 3GPP (Third Generation Partnership Project) standard as follows.

1) RAN domain includes mobile equipment (MEs), base stations (BSs) (*i.e.* eNodeB for outdoor, HeNodeB for indoor) where MEs are mobile equipment of 3GPP standard mobile devices and BSs forward messages from MEs to the serving network domain.

2) CN domain includes mobile management entities (MMEs) or serving gateways (S-GWs). An MME prepares services for the MEs's requests and S-GW forwards messages to another machines.

3) HN domain includes the Home facilitator server (HFS) which provides services for authentication process with MEs.

In the LTE networks, we assume that the network of service providers is secure. Data transmission between service providers' devices such as Home and MME is protected.

Table 1: The notations of entities in the network architecture

Notations	Definition
ME	Mobile Equipment (machine)
eNB	Type of base station (BS) called evolved Node B (eNodeB)
HeNB	Type of base station (BS) called Home evolved Node B (HeNodeB)
MME	Mobile Management Entity
S-GW	Serving Gateway
HFS	Home Facilitator Server

2.2 Elliptic Curve Cryptography

For the Elliptic Curve Cryptography (ECC), we describe the situation of Alice and Bob which they have a pair of keys (public key and private key) [18]. Public keys can be published. Alice and Bob can create a shared key for sending data in secure communication by using the Diffie-Hellman key exchange. The principle is as follows. In a finite field (Fq), an elliptic curve E is defined over Fq and P is a point on E (*i.e.* $P \in E$). Alice chooses a random secret a in Fq (*i.e.* $a \in Fq$) and computes her public key aP on E (*i.e.* $aP \in E$) and sends the key to Bob. In the same way, Bob chooses a random secret b and calculates bP on E and sends it to Alice. The secret common key between Alice and Bob is abP on E.

3 Security Analysis of SE-AKA Protocol

In this section, we give some security analysis of the SE-AKA for the LTE network. The SE-AKA protocol is used to facilitate mobile equipment (MEs) that have been subscribed to the home network (HN) to roam into a serving network (SN) which is far from HN. The SE-AKA protocol can be divided into 2 protocols:

1) Protocol execution for the first equipment;
2) Protocol execution for the remaining equipment of the same group. Because the supplier provides a group key (GK) to each group for secure communication, then all MEs of the group can know the group key. Table 2 shows the notations used in the SE-AKA protocol illustrated in Figure 2.

In the first device authentication process, the ME_1 uses a secret key which known only between it and the Home to generate a message authentication code (MAC) to authenticate itself with the Home via MME. Home verifies ME_1 by using the same secret key. If the verification is successful, the Home sends the group information management list (GIML), including group name, group ID, MEs' IDs and synchronization values (SVs) to MME/SN.

Table 2: Notations use in the SE-AKA protocol [8]

Notations	Definition
R_{G1-j}	The random number generated by ME_j in group G1
R_{MME}	The random number generated by MME
ID_{G1}	The identity of group G1
ID_{MME}	The identity of MME
$TID_{ME_{G1-j}}$	The temporary identity of ME_j in group G1
MAC_{MME}	The message authentication code computes by MME
$MAC_{ME_{G1-j}}$	The message authentication code computes ME_j in group G1
AMF	Authentication management field
LAI	Location Area Identification
$KGK_{ME_{G1-j}}$	The key generation key between ME_{G1-j} and MME
$f_{GTK_{G1}}$	A key generation function of group G1
aP, bP	A device's public key
abP	A shared key between two parties
ME	Mobile Equipment
MME	Mobile Management Entity
HSS	Home Subscriber Server

In self-confirmation of each remaining ME of the group, the GK and SV are mainly utilized in the authentication process. For GK, every ME knows this value and SV is not a key, so the security of this verification is reduced, and the authentication process can be easily attacked. Then, a group member can impersonate other ones who have not yet confirmed themselves.

As shown in Figure 2, an ME wants to disguise to be another one by sending the identity information ($AUTH_{ME_{G1-j}} = ID_{G1}||TID_{ME_{G1-j}}||R_{G1-j}$) of target member to the service MME. The MME uses a group temporary key (GTK) which got from Home (HSS) to perform mutual authentication with the ME without HSS's assistance. The GTK is generated from Home by using group key (GK). This key makes the MME to believe an ME.

In the protocol, the MME sends authentication request $AUTH_{MME} = (ID_{MME} \,||\, ID_{G1}||\, TID_{ME_{G1-j}}||\, MAC_{MME}||\, R_{HSS} \,||\, R_{MME} \,||\, R_{G1-j}||\, AMF \,||\, aP)$ where $MAC_{MME} = f_{GTK_{G1}}(ID_{MME}||\, ID_{G1}||\, TID_{ME_{G1-j}}||R_{HSS}||R_{MME}||R_{G1-j} ||AMF||aP||SV_{G1-j} + i)$ to the ME. The value i is the sequence of the mutual authentication with ME_{G1-j}. If the fake ME could ever attack the synchronization value (SV_{G1-j}), it selects a random number b and can computes bP, and computes $KGK_{ME_{G1-j}} = f_{GTK_{G1}}$ ($ID_{MME} \,||\, TID_{ME_{G1-j}} \,||\, R_{MME} \,||\, R_{G1-j} \,||\, abP$) and

$MAC_{ME_{G1-j}} = f_{KGK_{ME_{G1-j}}}(ID_{MME} || ID_{G1} || TID_{ME_{G1-j}} || R_{MME} || LAI || bp || abP || SV_{G1-j} + i)$. It then sends $(MAC_{ME_{G1-j}} || bp)$ to the MME.

Upon receiving the response, MME verifies $MAC_{ME_{G1-j}}$ by using the received information to compute $MAC_{ME_{G1-j}}$ by itself. It then compares the computed $MAC_{ME_{G1-j}}$ with the received $MAC_{ME_{G1-j}}$. If they are the same then MME believes that ME.

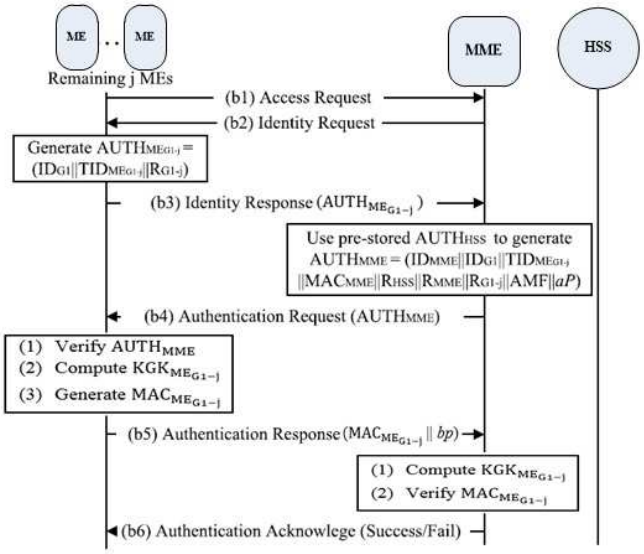


Figure 2: The authentication procedure of remaining MEs [8]

In this way, a member of the group will be able to disguise itself as other exist members. Although this protocol has some vulnerable points and is designed to work in the only one LTE network, the idea to seperate the authentication process into the authentication of the first device and the remaining devices can reduce the network traffic. According to this idea, we then apply it to create a new protocol.

4 The Proposed SE-GA Protocol

In this section, we propose SE-GA protocol for ME/MEs in a group to access into serving network domains. The design goals of SE-GA protocol are:

1) Members of the group must be independent.

2) The protocol allows the group in which members can come from different home networks and they can work on different networks at the same time as shown in Figure 3;

3) Each member cannot impersonate another member within the group;

4) Protocol must be able to prevent attacks such as secure key derivation, man-in-the-middle attacks, and so on. In addition, identity verification should be secure to ensure accuracy and to minimize interaction time.

4.1 Initialization

In the initial stage, each ME creates a pair of long-term private key and public key, and it sends the public key to its Home. Then the HN and ME can create a shared secret key by using a Diffie-Hellman key exchange. It is noted that a long-term public key of the Home is well-known. When several MEs form a group G_n, they create a session group key.

Each group member then sends the group's information, *i.e.* Group ID, number of members, Temporary identity numbers (TID) and all long-term public keys of the group members to his/her Home. This data is sent with integrity control by utilizing the shared key between the group member and the Home. The data does not need to be secret. However, if we need secrecy the information can be covered by using the shared key. On receiving the messages, each Home keeps the group's information in GDL as shown in Table 3.

Table 3: Group detail list (GDL)

Group number	Group ID	TID_{ME_i}	ID_{HFS_k}	Public key_{ME_i}
G_1	ID_{G_1}	TID_{ME_1}	ID_{HFS_1}	Pub_{ME_1}
.	.	TID_{ME_2}	ID_{HFS_2}	Pub_{ME_2}
.	.	.	.	.
.	.	.	.	.
.	.	TID_{ME_i}	ID_{HFS_k}	Pub_{ME_i}
G_2	ID_{G_2}	TID_{ME_1}	ID_{HFS_1}	Pub_{ME_1}
.	.	TID_{ME_3}	ID_{HFS_2}	Pub_{ME_3}
.	.	.	.	.
.	.	.	.	.
.	.	TID_{ME_m}	ID_{HFS_l}	Pub_{ME_m}

4.2 SE-GA Protocol for Each ME_i in a Group G_n

When an ME_i connects to a wireless point, it authenticates itself with that network in order to use network services.

In the authentication process, an ME_i device in a group G_n, connects to the wireless point in any area mobile management entity (MME_j). The ME_i then sends an access request $AUTH_i$ to the MME_j. When the MME_j receives a request, it checks whether the ME_i is a member in the previously requested group by using HFS_k and ID_{G_n} in the $AUTH_i$ to determine if a group detail list (GDL) exists in the MME_j's database. If not, ME_i is the first machine in the group that requests the connection with MME_j. MME_j then performs the authentication process for the first ME device (*i.e.* using case 1) and gets a GDL from ME_i's Home. Otherwise, if there is the GDL of that ME_i, then MME_j performs an authentication process as if the

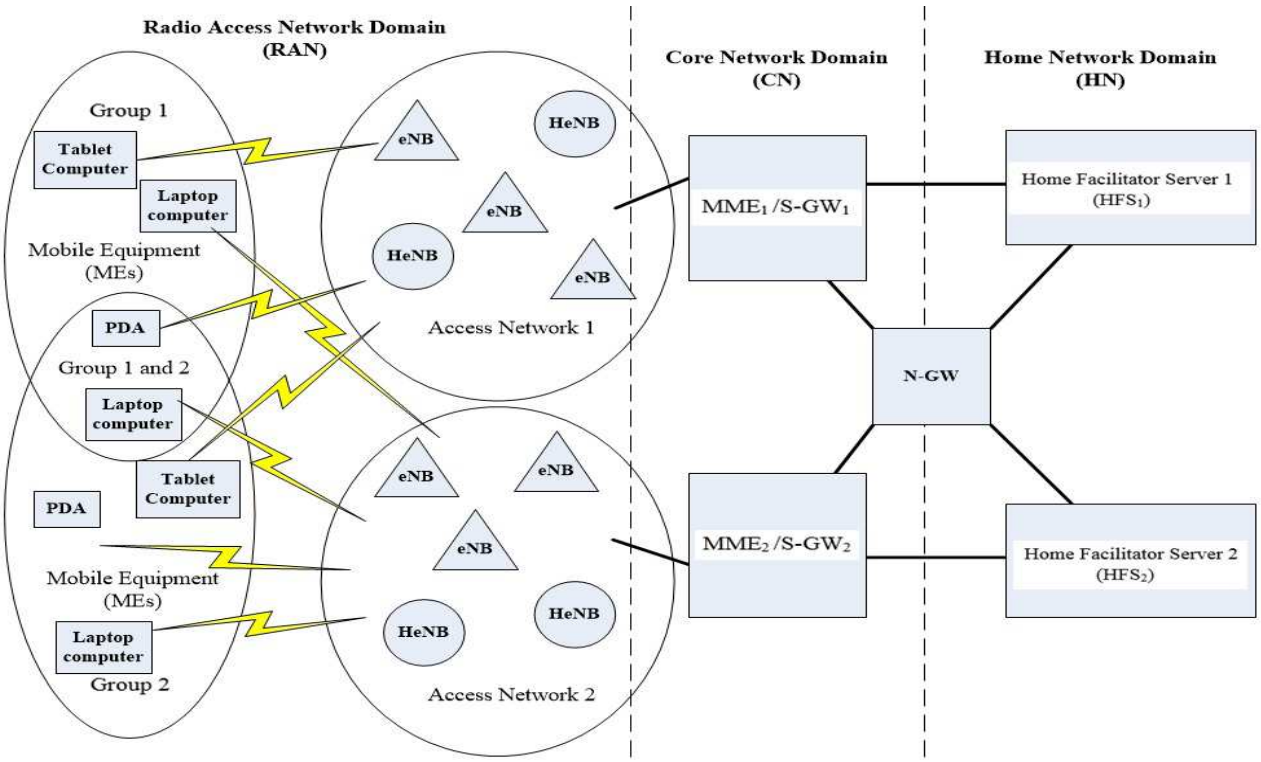


Figure 3: Network Architecture based on 3GPP standard in SE-GA protocol

ME_i is a remaining ME device (*i.e.* using case 2). Table 4 shows the notations used in the SE-GA protocol. The machine x or y can be an MME, HFS or ME. When x or y is represent by G_n- i, it means an ME_i of a group n.

The steps of the SE-GA protocol are as the following.

Case 1: Authentication for the first ME

If ME_i is the first member of a group G_n that want to authenticate with MME_j, then MME_j does not have a GDL of the ME_i's group in MME_j's database. Therefore, MME_j looks for the ME_i's home network (HFS_k) in the authentication request and then forwards the authentication data request, local area identification of MME_j, identity of MME_j and MAC_{MME_j} (*i.e.* $AUTH_i$, LAI_{MME_j}, ID_{MME_j}, MAC_{MME_j}) to HFS_k of ME_i through N-GW. If the authentication data request passes the network gateway (N-GW), the N-GW only forwards the authentication request to the destination (HFS_k). This case is composed of Steps 1 – 5 as shown in Figure 4.

Step 1. $ME_i \rightarrow MME_j$: **Access Request** ($AUTH_i$).

The ME_i generates $AUTH_i = (ID_{G_n} \,||\, TID_{ME_i} \,||\, R_{G_n-i} \,||\, TS_{G_n-i}||HFS_k||LAI_{ME_i}||bP||MAC_q||MAC_i)$ and sends it to MME_j. $MAC_q = f^1_{SK_{MME_j-ME_i}}(ID_{G_n} \,||\, TID_{ME_i} \,||\, R_{G_n-i} \,||\, TS_{G_n-i}||HFS_k||LAI_{ME_i}||bP)$ and it is used by MME_j to verify whether it is the correct ME_i. While $MAC_i = f^5_{SK_{ME_i-HFS_k}}(ID_{G_n} \,||\, TID_{ME_i} \,||\, R_{G_n-i} \,||\, TS_{G_n-i} \,||\, HFS_k \,||\, LAI_{ME_i} \,||\, bP)$ and it is used by HFS_k to verify whether it is the correct ME_i. The function $f^1_{SK_{MME_j-ME_i}}$ and $f^5_{SK_{ME_i-HFS_k}}$ are used for generating message authentication codes MAC_q and MAC_i respectively. $SK_{MME_j-ME_i}$ is a shared secret key between MME_j and ME_i, and is computed from ME_i's private key and MME_j's public key by using the Diffie-Hellman key exchange. It is noted that MME_j's public key is well-known on the internet. In part of $SK_{ME_i-HFS_k}$, it is a shared secret key between ME_i and its home network (HN) which is computed by performing the Diffie-Hellman key exchange in the initialization state. The value bP is a session public key of ME_i. It is created by selecting a random number b and computing bP on Elliptic Curve. TID_{ME_i} is a temporary identity of ME_i in HFS_k and is used for registration in 3GPP/LTE networks. The value is installed in ME_i by the supplier of ME_i.

Step 2. $MME_j \rightarrow HFS_k$: **Authentication Data Request** ($AUTH_i$, TS_{MME_j}, LAI_{MME_j}, ID_{MME_j}, MAC''_{MME_j}).

When the MME_j receives the authentication data request from ME_i, it uses HFS_k and ID_{G_n} in the $AUTH_i$ to find out whether this request is the first request of group, by searching for ID_{G_n} in the Group Detail List (GDL) of MME_j's database. If it cannot find the information in MME_j's database, then MME_j forwards $AUTH_i$, TS_{MME_j}, ID_{MME_j},

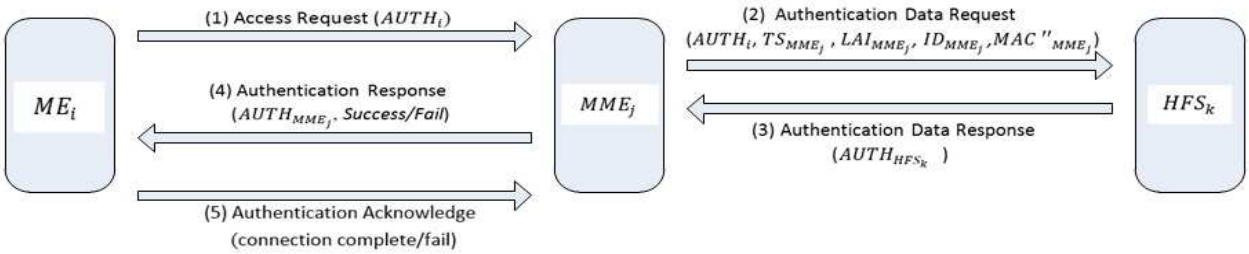


Figure 4: The SE-GA protocol for the first ME

$\mathrm{LAI_{MME_j}}$, $\mathrm{MAC''_{MME_j}}$ to the $\mathrm{HFS_k}$. The $\mathrm{LAI_{MME_j}}$ reports the location of the wireless point which $\mathrm{ME_i}$ connects to, and $\mathrm{MAC''_{MME_j}} = f^3_{\mathrm{SK_{MME_j-HFS_k}}}$ ($\mathrm{AUTH_i}$|| $\mathrm{TS_{MME_j}}$|| $\mathrm{ID_{MME_j}}$|| $\mathrm{LAI_{MME_j}}$). The long-term secret key ($\mathrm{SK_{MME_j-HFS_k}}$) between $\mathrm{MME_j}$ and $\mathrm{HFS_k}$ is computed by using the $\mathrm{HFS_k}$'s public key and $\mathrm{MME_j}$'s private key in the Diffie-Hellman key exchange. It is noted that $\mathrm{HFS_k}$'s public key is well-known on the internet.

Table 4: Notations used in the SE-GA protocol

Notations	Definition
$\mathrm{R_x}$	The random number generated by machine x
$\mathrm{TS_x}$	The time stamp generated by machine x
$\mathrm{ID_x}$	The identity of machine x
$\mathrm{PID_x}$	The permanent identity of machine x
$\mathrm{TID_x}$	The temporary identity of machine x
$\mathrm{SK_{x-y}}$	The shared secret key between machine x and y
$\mathrm{SSK_{x-y}}$	The shared session key between machine x and y
$\mathrm{MAC_x}$	The message authentication code computed by machine x
$\mathrm{LAI_x}$	Location Area Identification of machine x
$f^1_{\mathrm{SK_{MME_j-ME_i}}}$	MAC generating function using $\mathrm{SK_{MME_j-ME_i}}$
$f^2_{\mathrm{SK_{MME_j-ME_i}}}$	SSK generating function using $\mathrm{SK_{MME_j-ME_i}}$
$f^3_{\mathrm{SK_{MME_j-HFS_k}}}$	MAC generating function using $\mathrm{SK_{MME_j-HFS_k}}$
$f^4_{\mathrm{SK_{MME_j-HFS_k}}}$	MAC generating function using $\mathrm{SK_{MME_j-HFS_k}}$
$f^5_{\mathrm{SK_{ME_i-HFS_k}}}$	MAC generating function using $\mathrm{SK_{ME_i-HFS_k}}$
aP, bP	A device's public key
abP	A shared key between two parties

$\mathrm{MME_j}$ also keeps bP and $\mathrm{MAC_q}$ in order to use them afterward.

Step 3. $\mathrm{HFS_k} \rightarrow \mathrm{MME_j}$: **Authentication Data Response** ($\mathrm{AUTH_{HFS_k}}$).

Upon receiving authentication data request ($\mathrm{AUTH_i}$, $\mathrm{TS_{MME_j}}$, $\mathrm{LAI_{MME_j}}$, $\mathrm{ID_{MME_j}}$, $\mathrm{MAC''_{MME_j}}$) from $\mathrm{MME_j}$, the $\mathrm{HFS_k}$ verifies $\mathrm{MME_j}$ by computing $\mathrm{MAC'''_{MME_j}} = f^3_{\mathrm{SK_{MME_j-HFS_k}}}$ ($\mathrm{AUTH_i}$||$\mathrm{TS_{MME_j}}$||$\mathrm{ID_{MME_j}}$||$\mathrm{LAI_{MME_j}}$) and compares it with $\mathrm{MAC''_{MME_j}}$. Here, $\mathrm{SK_{MME_j-HFS_k}}$ is computed by using $\mathrm{HFS_k}$'s private key and $\mathrm{MME_j}$'s public key. If it is the same MAC value then $\mathrm{HFS_k}$ believes that the message is sent from $\mathrm{MME_j}$.

Before $\mathrm{HFS_k}$ verifies $\mathrm{MAC_i}$ which is in $\mathrm{AUTH_i}$, the $\mathrm{HFS_k}$ compares $\mathrm{LAI_{MME_j}}$ with $\mathrm{LAI_{ME_i}}$ to check whether they are the same. If they have the same value, $\mathrm{HFS_k}$ verifies $\mathrm{MAC_i}$ by computing $\mathrm{MAC'_i} = f^5_{\mathrm{SK_{ME_i-HFS_k}}}$ ($\mathrm{ID_{G_n}}$||$\mathrm{TID_{ME_i}}$||$\mathrm{R_{G_n-i}}$||$\mathrm{TS_{G_n-i}}$||$\mathrm{HFS_k}$||$\mathrm{LAI_{ME_i}}$||bP) from data in $\mathrm{AUTH_i}$. Then $\mathrm{HFS_k}$ compares $\mathrm{MAC'_i}$ with the $\mathrm{MAC_i}$. If these values are the same, the $\mathrm{HFS_k}$ can believe that the message is sent from $\mathrm{ME_i}$.

The $\mathrm{HFS_k}$ then generates $\mathrm{AUTH_{HFS_k}} = (\mathrm{R_{G_n-i}}||$ $\mathrm{ID_{HFS_k}}$ || $\mathrm{HFS_k}$ || GDL || $\mathrm{TS_{HFS_k}}$ || $\mathrm{MAC_{HFS_k}}$), where $\mathrm{MAC_{HFS_k}} = f^4_{\mathrm{SK_{MME_j-HFS_k}}}$ ($\mathrm{R_{G_n-i}}$ || $\mathrm{ID_{HFS_k}}$ || $\mathrm{HFS_k}$ || GDL || $\mathrm{TS_{HFS_k}}$) and it sends $\mathrm{AUTH_{HFS_k}}$ to the $\mathrm{MME_j}$. GDL is composed of group number, group identity, temporary identity of every $\mathrm{ME_i}$, identity of $\mathrm{HFS_k}$ and public keys of all MEs in this group as shown in Table 2.

Step 4. $\mathrm{MME_j} \rightarrow \mathrm{ME_i}$: **Authentication Response** ($\mathrm{AUTH_{MME_j}}$, Success/Fail).

After $\mathrm{MME_j}$ receives $\mathrm{AUTH_{HFS_k}}$ from $\mathrm{HFS_k}$, $\mathrm{MME_j}$ computes $\mathrm{MAC'_{HFS_k}} = f^4_{\mathrm{SK_{MME_j-HFS_k}}}$ ($\mathrm{R_{G_n-i}}$ || $\mathrm{ID_{HFS_k}}$ || $\mathrm{HFS_k}$||GDL||$\mathrm{TS_{HFS_k}}$) to verify the message from $\mathrm{HFS_k}$. If the verification passes, $\mathrm{MME_j}$ computes $\mathrm{MAC'_q} = f^1_{\mathrm{SK_{MME_j-ME_i}}}$ ($\mathrm{ID_{G_n}}$ || $\mathrm{TID_{ME_i}}$ || $\mathrm{R_{G_n-i}}$ || $\mathrm{TS_{G_n-i}}$|| $\mathrm{HFS_k}$|| $\mathrm{LAI_{ME_i}}$||bP) and compares it with $\mathrm{MAC_q}$ from Step 1. The $\mathrm{SK_{MME_j-ME_i}}$ is computed by $\mathrm{MME_j}$'s

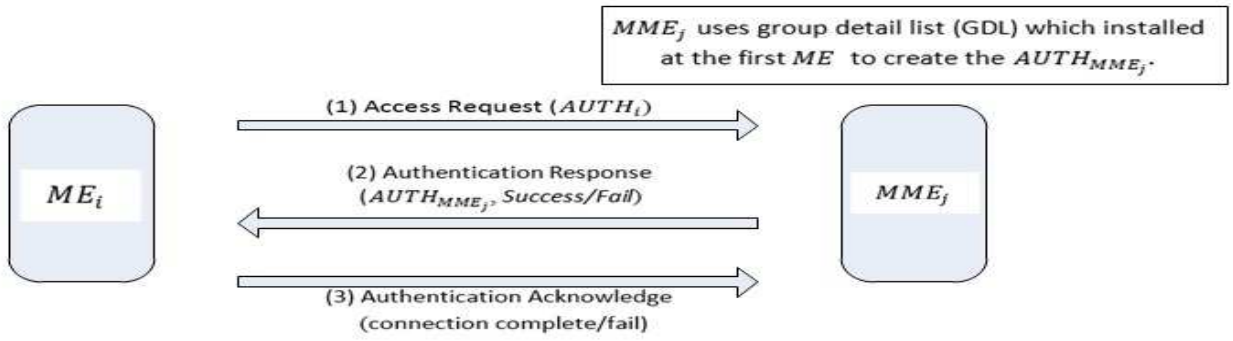


Figure 5: The SE-GA protocol for remaining ME devices

private key and ME_i's long-term public key got from GDL. If $MAC'_q = MAC_q$, MME_j installs GDL of G_n into MME_j's database. The GDL facilitates the MME_j to check the remaining ME_i's authentication information. Then, MME_j can trust the message $AUTH_i$ which is sent by ME_i, because MME_j got correct response from ME_i's Home.

MME_j then randomizes a number a to compute a session public key aP and a secret value abP on Elliptic Curve. Note that bP is obtained from Step 1. MME_j also generates $AUTH_{MME_j} = (ID_{MME_j}||ID_{G_n}||TID_{ME_i}||R_{MME_j}$ $||R_{G_n-i}||\ TS'_{MME_j}||\ aP||MAC_{MME_j})$, where $MAC_{MME_j} = f^1_{SK_{MME_j-ME_i}}$ $(ID_{MME_j}\ ||\ ID_{G_n}\ ||$ $TID_{ME_i}\ ||\ R_{MME_j}\ ||\ R_{G_n-i}\ ||\ TS'_{MME_j}||aP)$. It then sends $AUTH_{MME_j}$ and a response 'success' to ME_i. MME_j can now compute session key between it and ME_i by $SSK_{MME_j-ME_i} = f^2_{SK_{MME_j-ME_i}}$ $(ID_{MME_j}||TID_{ME_i}||R_{MME_j}||R_{G_n-i}||abP)$.

Step 5. $ME_i \rightarrow MME_j$: **Authentication Acknowledge** (connection complete/fail).

When the ME_i gets the authentication data response from MME_j, it verifies MME_j by computing $MAC'_{MME_j} = f^1_{SK_{MME_j-ME_i}}$ $(ID_{MME_j}\ ||\ ID_{G_n}$ $||\ TID_{ME_i}\ ||\ R_{MME_j}\ ||\ R_{G_n}-i\ ||\ TS'_{MME_j}\ ||\ aP)$ and compares MAC_{MME_j} with MAC'_{MME_j}. The $SK_{MME_j-ME_i}$ is computed from ME_i's private key and MME_j's public key by using the Diffie-Hellman key exchange. MME_j's long-term public key is well-known on the internet.

If MAC_{MME_j} and MAC'_{MME_j} are the same then it is the correct MME_j. ME_i then computes abP by using aP from $AUTH_{MME_j}$ and creates a session key between ME_i and MME_j by $SSK_{MME_j-ME_i} = f^2_{SK_{MME_j-ME_i}}$ $(ID_{MME_j}||TID_{ME_i}||R_{MME_j}||R_{G_n-i}||abP)$. Now, the ME_i has a shared session key $SSK_{MME_j-ME_i}$ with MME_j and sends connection complete to MME_j. Otherwise, ME_i sends a response, 'connection failure' to MME_j.

Case 2: Authentication for the remaining MEs

If ME_i is a remaining member of the group G_n that has a member authenticated with MME_j, then MME_j has the group detail list (GDL) of group G_n in the MME_j's database. The MME_j can use the ME_i's public key in GDL to create a shared secret key ($SK_{MME_j-ME_i}$) between MME_j and ME_i. This case is composed of Steps 1 – 3 as shown in Figure 5.

Step 1. $ME_i \rightarrow MME_j$: **Access Request** ($AUTH_i$).

The ME_i generates $AUTH_i = (ID_{G_n}\ ||\ TID_{ME_i}\ ||$ $R_{G_n}-i\ ||\ TS_{G_n-i}\ ||\ HFS_k\ ||\ LAI_{ME_i}\ ||\ bP\ ||\ MAC_q$ $||\ MAC_i)$, $MAC_q = f^1_{SK_{MME_j-ME_i}}$ $(ID_{G_n}\ ||\ TID_{ME_i}\ ||$ $R_{G_n-i}\ ||\ TS_{G_n-i}\ ||\ HFS_k\ ||\ LAI_{ME_i}\ ||\ bP)$ and MAC_i $= f^5_{SK_{ME_i-HFS_k}}$ $(ID_{G_n}\ ||\ TID_{ME_i}\ ||\ R_{G_n-i}\ ||\ TS_{G_n-i}\ ||$ $HFS_k\ ||\ LAI_{ME_i}\ ||\ bP)$ and sends $AUTH_i$ to MME_j.

Step 2. $MME_j \rightarrow ME_i$: **Authentication Response** ($AUTH_{MME_j}$, Success/Fail).

When the MME_j receives an authentication data request from ME_i, it checks the request of ME_i by using HFS_k and ID_{G_n} in the $AUTH_i$ to find out whether this request is the first request of group, by searching for ID_{G_n} in the Group Detail List (GDL) of MME_j's database. If it can find ID_{G_n}, then MME_j computes a long-term secret key ($SK_{MME_j} - ME_i$) between MME_j and ME_i by using ME_i's public key in GDL and MME_j's private key.

Before MME_j verifies MAC_q which is in $AUTH_i$, the MME_j compares LAI_{ME_i} with LAI_{MME_j} to check whether they are the same. If they have the same value, the MME_j computes MAC'_q $= f^1_{SK_{MME_j-ME_i}}$ $(ID_{G_n}||TID_{ME_i}||R_{G_n-i}||TS_{G_n-i}$ $||HFS_k||LAI_{ME_i}||bP)$. It then compares MAC'_q with MAC_q from Step (1). If $MAC'_q = MAC_q$ then MME_j trusts ME_i and messages are sent by ME_i.

MME_j then randomizes a number a to compute a session public key aP and a secret value abP on Elliptic Curve. Further, MME_j generates $AUTH_{MME_j}$

$= (\mathrm{ID_{MME_j}} \,||\, \mathrm{ID_{G_n}} \,||\, \mathrm{TID_{ME_i}} \,||\, \mathrm{R_{MME_j}} \,||\, \mathrm{R_{G_n-i}} \,||\, \mathrm{TS'_{MME_j}} \,||\, aP \,||\, \mathrm{MAC_{MME_j}})$, where $\mathrm{MAC_{MME_j}} = f^1_{\mathrm{SK_{MME_j-ME_i}}}(\mathrm{ID_{MME_j}} \,||\, \mathrm{ID_{G_n}} \,||\, \mathrm{TID_{ME_i}} \,||\, \mathrm{R_{MME_j}} \,||\, \mathrm{R_{G_n-i}} \,||\, \mathrm{TS'_{MME_j}} \,||\, aP)$. It then sends $\mathrm{AUTH_{MME_j}}$ and a response, 'success' to $\mathrm{ME_i}$.

Now, $\mathrm{MME_j}$ can compute a session key between it and $\mathrm{ME_i}$ by $\mathrm{SSK_{MME_j-ME_i}} = f^2_{\mathrm{SK_{MME_j-ME_i}}}(\mathrm{ID_{MME_j}}||\mathrm{TID_{ME_i}}||\mathrm{R_{MME_j}}||\mathrm{R_{G_n-i}}||abP)$.

Step 3. $\mathrm{ME_i} \rightarrow \mathrm{MME_j}$: **Authentication Acknowledge** (connection complete/fail).

When the $\mathrm{ME_i}$ gets the authentication response from $\mathrm{MME_j}$, it verifies the message by computing $\mathrm{MAC'_{MME_j}} = f^1_{\mathrm{SK_{MME_j-ME_i}}}(\mathrm{ID_{MME_j}}||\mathrm{ID_{G_n}}||\mathrm{TID_{ME_i}}||\mathrm{R_{MME_j}}||\mathrm{R_{G_n-i}}||\mathrm{TS'_{MME_j}}||aP)$ and compares $\mathrm{MAC_{MME_j}}$ with $\mathrm{MAC'_{MME_j}}$. The $\mathrm{SK_{MME_j-ME_i}}$ is computed from $\mathrm{ME_i}$'s private key and $\mathrm{MME_j}$'s public key.

If $\mathrm{MAC_{MME_j}}$ and $\mathrm{MAC'_{MME_j}}$ have the same value then $\mathrm{ME_i}$ believes that the message is sent from $\mathrm{MME_j}$. $\mathrm{ME_i}$ then computes abP by using aP from $\mathrm{AUTH_{MME_j}}$ and creates session key between $\mathrm{ME_i}$ and $\mathrm{MME_j}$ by $\mathrm{SSK_{MME_j-ME_i}} = f^2_{\mathrm{SK_{MME_j-ME_i}}}(\mathrm{ID_{MME_j}}||\mathrm{TID_{ME_i}}||\mathrm{R_{MME_j}}||\mathrm{R_{G_n-i}}||abP)$. Now, the $\mathrm{ME_i}$ has a shared session key $\mathrm{SSK_{MME_j-ME_i}}$ with $\mathrm{MME_j}$ and sends connection complete to $\mathrm{MME_j}$. Otherwise, if $\mathrm{MAC_{MME_j}}$ and $\mathrm{MAC'_{MME_j}}$ are not the same then $\mathrm{ME_i}$ sends a response, 'connection failure' to $\mathrm{MME_j}$.

5 Authentication Proof by using BAN Logic

In this section, we give a proof of the SE-GA protocol by using the well-known BAN Logic. The notations used in SE-GA protocol are listed in Table 5.

Table 5: Notations used in the proof

Notations	Definition
bP	A session public key of $\mathrm{ME_i}$
aP	A session public key of $\mathrm{MME_j}$
$\mathrm{SK_{ME_i-HFS_k}}$	A long-term secret shared between $\mathrm{ME_i}$ and $\mathrm{HFS_k}$
$\mathrm{SK_{ME_i-MME_j}}$	A long-term secret shared between $\mathrm{ME_i}$ and $\mathrm{MME_j}$
$\mathrm{SK_{MME_j-HFS_k}}$	A long-term secret shared between $\mathrm{MME_j}$ and $\mathrm{HFS_k}$
$\mathrm{SSK_{MME_j-ME_i}}$	A shared session key between $\mathrm{MME_j}$ and $\mathrm{ME_i}$

We will prove the authentication of the mobile equipment in both cases: the case of the first ME device and the case of the remaining ME devices.

5.1 Authentication Proof for the First ME

The communicating messages used in the case of the first ME device are as follows:

(a) $\mathrm{ME_i} \rightarrow \mathrm{MME_j}$:
$\mathrm{AUTH_i} = (\mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{TS_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP, \mathrm{MAC_q}((\mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{TS_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP), \mathrm{SK_{ME_i-MME_j}}), \mathrm{MAC_i}((\mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{TS_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP), \mathrm{SK_{ME_i-HFS_k}}))$.

(b) $\mathrm{MME_j} \rightarrow \mathrm{HFS_k}$:
$(\mathrm{AUTH_i}, \mathrm{TS_{MME_j}}, \mathrm{LAI_{MME_j}}, \mathrm{ID_{MME_j}}, \mathrm{MAC''_{MME_j}}((\mathrm{AUTH_i}, \mathrm{TS_{MME_j}}, \mathrm{LAI_{MME_j}}, \mathrm{ID_{MME_j}}), \mathrm{SK_{MME_j-HFS_k}}))$.

(c) $\mathrm{HFS_k} \rightarrow \mathrm{MME_j}$:
$(\mathrm{R_{G_n-i}}, \mathrm{ID_{HFS_k}}, \mathrm{HFS_k}, \mathrm{GDL}, \mathrm{TS_{HFS_k}}), \mathrm{MAC_{HFS_k}}((\mathrm{R_{G_n-i}}, \mathrm{ID_{HFS_k}}, \mathrm{HFS_k}, \mathrm{GDL}, \mathrm{TS_{HFS_k}}), \mathrm{SK_{MME_j-HFS_k}})$.

(d) $\mathrm{MME_j} \rightarrow \mathrm{ME_i}$:
$(\mathrm{ID_{MME_j}}, \mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{MME_j}}, \mathrm{R_{G_n-i}}, \mathrm{TS'_{MME_j}}, aP), \mathrm{MAC_{MME_j}}((\mathrm{ID_{MME_j}}, \mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{MME_j}}, \mathrm{R_{G_n-i}}, \mathrm{TS'_{MME_j}}, aP), \mathrm{SK_{ME_i-MME_j}})$.

The messages can be transformed into the idealized forms as

(a) $\mathrm{ME_i} \rightarrow \mathrm{MME_j}$:
$\mathrm{AUTH_i} = <\mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{TS_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP >_{\mathrm{SK_{ME_i-MME_j}}} <\mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{TS_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP >_{\mathrm{SK_{ME_i-HFS_k}}}$

(b) $\mathrm{MME_j} \rightarrow \mathrm{HFS_k}$:
$<\mathrm{AUTH_i}, \mathrm{LAI_{MME_j}}, \mathrm{TS_{MME_j}}, \mathrm{ID_{MME_j}} >_{\mathrm{SK_{MME_j-HFS_k}}}$

(c) $\mathrm{HFS_k} \rightarrow \mathrm{MME_j}$:
$<\mathrm{R_{G_n-i}}, \mathrm{ID_{HFS_k}}, \mathrm{HFS_k}, \mathrm{GDL}, \mathrm{TS_{HFS_k}} >_{\mathrm{SK_{MME_j-HFS_k}}}$

(d) $\mathrm{MME_j} \rightarrow \mathrm{ME_i}$:
$<\mathrm{ID_{MME_j}}, \mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{MME_j}}, \mathrm{R_{G_n-i}}, \mathrm{TS'_{MME_j}}, aP >_{\mathrm{SK_{ME_i-MME_j}}}$

In this form $\mathrm{TS_{G_n-i}}, \mathrm{TS_{MME_j}}, \mathrm{TS'_{MME_j}}, \mathrm{TS_{HFS_k}}$ **are nonces.**

We need to prove that $\mathrm{MME_j}$ believes $\mathrm{ME_i}$'s long term public key in GDL which it has received from $\mathrm{HFS_k}$ and uses the key to compute a long-term secret key ($\mathrm{SK_{ME_i-MME_j}}$) between $\mathrm{MME_j}$ and $\mathrm{ME_i}$. $\mathrm{MME_j}$ uses $\mathrm{SK_{ME_i-MME_j}}$ to verify $\mathrm{ME_i}$'s message. It then can believe $\mathrm{ME_i}$'s session public key, bP. Further, it needs to prove that $\mathrm{ME_i}$ can believe $\mathrm{MME_j}$'s session public key, aP. Both $\mathrm{MME_j}$ and $\mathrm{ME_i}$ can use aP and bP to compute a shared session secret, abP. To analyze this protocol, the following assumptions are made.

(1) $\mathrm{HFS_k}$ believes $\mathrm{MME_j} \overset{\mathrm{SK_{MME_j-HFS_k}}}{\longleftrightarrow} \mathrm{HFS_k}$.
(2) $\mathrm{HFS_k}$ believes $\mathrm{ME_i} \overset{\mathrm{SK_{ME_i-HFS_k}}}{\longleftrightarrow} \mathrm{HFS_k}$.

(3) $\mathrm{MME_j}$ believes $\mathrm{HFS_k} \overset{\mathrm{SK_{MME_j-HFS_k}}}{\longleftrightarrow} \mathrm{MME_j}$.
(4) $\mathrm{ME_i}$ believes $\mathrm{MME_j} \overset{\mathrm{SK_{ME_i-MME_j}}}{\longleftrightarrow} \mathrm{ME_i}$.
(5) $\mathrm{MME_j}$ believes fresh $(\mathrm{TS_{G_n-i}})$.
(6) $\mathrm{MME_j}$ believes fresh $(\mathrm{TS_{HFS_k}})$.
(7) $\mathrm{HFS_k}$ believes fresh $(\mathrm{TS_{G_n-i}})$.
(8) $\mathrm{HFS_k}$ believes fresh $(\mathrm{TS_{MME_j}})$.
(9) $\mathrm{ME_i}$ believes fresh $(\mathrm{TS'_{MME_j}})$.
(10) $\mathrm{HFS_k}$ believes $\mathrm{MME_j}$ control $(\mathrm{AUTH_i}, \mathrm{TS_{MME_j}}, \mathrm{LAI_{MME_j}}, \mathrm{ID_{MME_j}})$.
(11) $\mathrm{HFS_k}$ believes $\mathrm{ME_i}$ control $(\mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP)$.
(12) $\mathrm{MME_j}$ believes $\mathrm{HFS_k}$ controls $(\mathrm{R_{G_n-i}}, \mathrm{ID_{HFS_k}}, \mathrm{HFS_k}, \mathrm{GDL})$.
(13) $\mathrm{MME_j}$ believes $\mathrm{ME_i}$ controls $(\mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP)$.
(14) $\mathrm{ME_i}$ believes $\mathrm{MME_j}$ controls $(\mathrm{ID_{MME_j}}, \mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{MME_j}}, \mathrm{R_{G_n-i}}, aP)$.

The steps of the proof are as follows:

a) $\mathrm{HFS_k}$ believes $\mathrm{MME_j} \overset{\mathrm{SK_{MME_j-HFS_k}}}{\longleftrightarrow} \mathrm{HFS_k}$
and $\mathrm{HFS_k}$ sees $< \mathrm{AUTH_i}, \mathrm{LAI_{MME_j}}, \mathrm{TS_{MME_j}}, \mathrm{ID_{MME_j}} >_{\mathrm{SK_{MME_j-HFS_k}}}$,
then $\mathrm{HFS_k}$ believes $\mathrm{MME_j}$ said $(\mathrm{AUTH_i}, \mathrm{LAI_{MME_j}}, \mathrm{TS_{MME_j}}, \mathrm{ID_{MME_j}})$.

b) $\mathrm{HFS_k}$ believes fresh $(\mathrm{TS_{MME_j}})$
and $\mathrm{HFS_k}$ believes $\mathrm{MME_j}$ said $(\mathrm{AUTH_i}, \mathrm{LAI_{MME_j}}, \mathrm{TS_{MME_j}}, \mathrm{ID_{MME_j}})$,
then $\mathrm{HFS_k}$ believes $\mathrm{MME_j}$ believes $(\mathrm{AUTH_i}, \mathrm{LAI_{MME_j}}, \mathrm{TS_{MME_j}}, \mathrm{ID_{MME_j}})$.

The conjunction can be broken and the result is $\mathrm{HFS_k}$ believes $\mathrm{MME_j}$ believes $(\mathrm{AUTH_i}, \mathrm{LAI_{MME_j}}, \mathrm{ID_{MME_j}})$.

c) $\mathrm{HFS_k}$ believes $\mathrm{MME_j}$ control $(\mathrm{AUTH_i}, \mathrm{LAI_{MME_j}}, \mathrm{ID_{MME_j}})$
and $\mathrm{HFS_k}$ believes $\mathrm{MME_j}$ believes $(\mathrm{AUTH_i}, \mathrm{LAI_{MME_j}}, \mathrm{ID_{MME_j}})$,
then $\mathrm{HFS_k}$ believes $(\mathrm{AUTH_i}, \mathrm{LAI_{MME_j}}, \mathrm{ID_{MME_j}})$.

In steps *a) - c)*, $\mathrm{HFS_k}$ uses a long-term secret key between $\mathrm{MME_j}$ and $\mathrm{HFS_k}$ (*i.e.* $\mathrm{SK_{MME_j-HFS_k}}$) to verify the message $(\mathrm{AUTH_i}, \mathrm{LAI_{MME_j}}, \mathrm{TS_{MME_j}}, \mathrm{ID_{MME_j}})$ received from $\mathrm{MME_j}$. If the verification passes, $\mathrm{HFS_k}$ believes that the message is sent from $\mathrm{MME_j}$.

After $\mathrm{HFS_k}$ believes the message is sent from $\mathrm{MME_j}$, it verifies the authentication message $(< \mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{TS_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP >_{\mathrm{SK_{ME_i-HFS_k}}})$ which is in $\mathrm{AUTH_i}$. If the verification passes, $\mathrm{HFS_k}$ believes that the message is from $\mathrm{ME_i}$. The proof is as follows.

d) $\mathrm{HFS_k}$ believes $\mathrm{ME_i} \overset{\mathrm{SK_{ME_i-HFS_k}}}{\longleftrightarrow} \mathrm{HFS_k}$ **and** $\mathrm{HFS_k}$ sees $< \mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{TS_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP >_{\mathrm{SK_{ME_i-HFS_k}}}$),
then $\mathrm{HFS_k}$ believes $\mathrm{ME_i}$ said $(\mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{TS_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP)$.

e) $\mathrm{HFS_k}$ believes fresh $(\mathrm{TS_{G_n-i}})$ **and**
$\mathrm{HFS_k}$ believes $\mathrm{ME_i}$ said $(\mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{TS_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP)$,
then $\mathrm{HFS_k}$ believes $\mathrm{ME_i}$ believes $(\mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{TS_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP)$.

The conjunction can be broken and the result is $\mathrm{HFS_k}$ believes $\mathrm{ME_i}$ believes $(\mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP)$.

f) $\mathrm{HFS_k}$ believes $\mathrm{ME_i}$ control $(\mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP)$
and $\mathrm{HFS_k}$ believes $\mathrm{ME_i}$ believes $(\mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP)$,
then $\mathrm{HFS_k}$ believes $(\mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP)$.

In steps *d) - f)*, $\mathrm{HFS_k}$ verifies message $\mathrm{MAC_i}$ $(\mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{TS_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP)$ by computing $\mathrm{MAC'_i}$. The $\mathrm{HFS_k}$ then compares $\mathrm{MAC'_i}$ with the $\mathrm{MAC_i}$. If the verification passes, it is the correct $\mathrm{ME_i}$. Then $\mathrm{HFS_k}$ believes authentication message from $\mathrm{ME_i}$.

After that, $\mathrm{HFS_k}$ sends the authentication message $(\mathrm{R_{G_n-i}}, \mathrm{ID_{HFS_k}}, \mathrm{HFS_k}, \mathrm{GDL}, \mathrm{TS_{HFS_k}})$ to $\mathrm{MME_j}$.

g) $\mathrm{MME_j}$ believes $\mathrm{HFS_k} \overset{\mathrm{SK_{MME_j-HFS_k}}}{\longleftrightarrow} \mathrm{MME_j}$
and $\mathrm{MME_j}$ sees $< \mathrm{R_{G_n-i}}, \mathrm{ID_{HFS_k}}, \mathrm{HFS_k}, \mathrm{GDL}, \mathrm{TS_{HFS_k}} >_{\mathrm{SK_{MME_j-HFS_k}}}$,
then $\mathrm{MME_j}$ believes $\mathrm{HFS_k}$ said $(\mathrm{R_{G_n-i}}, \mathrm{ID_{HFS_k}}, \mathrm{HFS_k}, \mathrm{GDL}, \mathrm{TS_{HFS_k}})$.

h) $\mathrm{MME_j}$ believes fresh $(\mathrm{TS_{HFS_k}})$
and $\mathrm{MME_j}$ believes $\mathrm{HFS_k}$ said $(\mathrm{R_{G_n-i}}, \mathrm{ID_{HFS_k}}, \mathrm{HFS_k}, \mathrm{GDL}, \mathrm{TS_{HFS_k}})$,
then $\mathrm{MME_j}$ believes $\mathrm{HFS_k}$ believes $(\mathrm{R_{G_n-i}}, \mathrm{ID_{HFS_k}}, \mathrm{HFS_k}, \mathrm{GDL}, \mathrm{TS_{HFS_k}})$.

The conjunction can be broken and the result is $\mathrm{MME_j}$ believes $\mathrm{HFS_k}$ believes $(\mathrm{R_{G_n-i}}, \mathrm{ID_{HFS_k}}, \mathrm{HFS_k}, \mathrm{GDL})$.

i) $\mathrm{MME_j}$ believes $\mathrm{HFS_k}$ controls $(\mathrm{R_{G_n-i}}, \mathrm{ID_{HFS_k}}, \mathrm{HFS_k}, \mathrm{GDL})$
and $\mathrm{MME_j}$ believes $\mathrm{HFS_k}$ believes $(\mathrm{R_{G_n-i}}, \mathrm{ID_{HFS_k}}, \mathrm{HFS_k}, \mathrm{GDL})$,
then $\mathrm{MME_j}$ believes $(\mathrm{R_{G_n-i}}, \mathrm{ID_{HFS_k}}, \mathrm{HFS_k}, \mathrm{GDL})$.

In steps *g) – i)*, $\mathrm{MME_j}$ gets message $(\mathrm{R_{G_n-i}}, \mathrm{ID_{HFS_k}}, \mathrm{HFS_k}, \mathrm{GDL}, < \mathrm{R_{G_n-i}}, \mathrm{ID_{HFS_k}}, \mathrm{HFS_k}, \mathrm{GDL} >_{\mathrm{SK_{MME_j-HFS_k}}})$ from $\mathrm{HFS_k}$, and uses a long-term secret key $(\mathrm{SK_{MME_j-HFS_k}})$ between $\mathrm{MME_j}$ and $\mathrm{HFS_k}$ to verify message from $\mathrm{HFS_k}$. If the verification passes, $\mathrm{MME_j}$ believes that the message is from $\mathrm{HFS_k}$.

After that, $\mathrm{MME_j}$ verifies the authentication message $\mathrm{MAC_q}$ from $\mathrm{ME_i}$ as follows.

j) $\mathrm{MME_j}$ believes $\mathrm{ME_i} \overset{\mathrm{SK_{ME_i-MME_j}}}{\longleftrightarrow} \mathrm{MME_j}$
and $\mathrm{MME_j}$ sees $< \mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{TS_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP >_{\mathrm{SK_{ME_i-MME_j}}}$,
then $\mathrm{MME_j}$ believes $\mathrm{ME_i}$ said
$(\mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{TS_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP)$.

k) $\mathrm{MME_j}$ believes fresh $(\mathrm{TS_{G_n-i}})$ **and** $\mathrm{MME_j}$ believes $\mathrm{ME_i}$ said $(\mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{TS_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP)$,
then $\mathrm{MME_j}$ believes $\mathrm{ME_i}$ believes
$(\mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{TS_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP)$.

The conjunction can be broken and the result is $\mathrm{MME_j}$ believes $\mathrm{ME_i}$ believes $(\mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP)$.

l) $\mathrm{MME_j}$ believes $\mathrm{ME_i}$ controls
$(\mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}), bP)$
and $\mathrm{MME_j}$ believe $\mathrm{ME_i}$ believes
$(\mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP)$,
then $\mathrm{MME_j}$ believes
$(\mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP)$.

In steps *j) - l)*, $\mathrm{MME_j}$ verifies message $\mathrm{MAC_q}$ $(\mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{TS_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP)$ from $\mathrm{ME_i}$ by using $\mathrm{SK_{ME_i-MME_j}}$ to compute $\mathrm{MAC'_q}$. If the verification passes, it is the correct $\mathrm{ME_i}$. Then $\mathrm{MME_j}$ believes authentication message from $\mathrm{ME_i}$.

After that, $\mathrm{MME_j}$ selects random number a and computes aP and uses bP in $\mathrm{ME_i}$'s message to compute abP. $\mathrm{MME_j}$ now can compute a shared session key $\mathrm{SSK_{MME_j-ME_i}}$ between $\mathrm{MME_j}$ and $\mathrm{ME_i}$. $\mathrm{MME_j}$ then sends the authentication message $\mathrm{MAC_{MME_j}}$ $(\mathrm{ID_{MME_j}}, \mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{MME_j}}, \mathrm{R_{G_n-i}}, \mathrm{TS'_{MME_j}}, aP)$ to $\mathrm{ME_i}$.

m) $\mathrm{ME_i}$ believes $\mathrm{MME_j} \overset{\mathrm{SK_{ME_i-MME_j}}}{\longleftrightarrow} \mathrm{ME_i}$ **and** $\mathrm{ME_i}$ sees $< \mathrm{ID_{MME_j}}, \mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{MME_j}}, \mathrm{R_{G_n-i}}, \mathrm{TS'_{MME_j}}, aP >_{\mathrm{SK_{ME_i-MME_j}}}$,
then $\mathrm{ME_i}$ believes $\mathrm{MME_j}$ said
$(\mathrm{ID_{MME_j}}, \mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{MME_j}}, \mathrm{R_{G_n-i}}, \mathrm{TS'_{MME_j}}, aP)$.

n) $\mathrm{ME_i}$ believes fresh $(\mathrm{TS'_{MME_j}})$ **and** $\mathrm{ME_i}$ believes $\mathrm{MME_j}$ said $(\mathrm{ID_{MME_j}}, \mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{MME_j}}, \mathrm{R_{G_n-i}}, \mathrm{TS'_{MME_j}}, aP)$,
then $\mathrm{ME_i}$ believes $\mathrm{MME_j}$ believes
$(\mathrm{ID_{MME_j}}, \mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{MME_j}}, \mathrm{R_{G_n-i}}, \mathrm{TS'_{MME_j}}, aP)$.

The conjunction can be broken and the result is $\mathrm{ME_i}$ believes $\mathrm{MME_j}$ believes $(\mathrm{ID_{MME_j}}, \mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{MME_j}}, \mathrm{R_{G_n-i}}, aP)$.

o) $\mathrm{ME_i}$ believes $\mathrm{MME_j}$ controls
$(\mathrm{ID_{MME_j}}, \mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{MME_j}}, \mathrm{R_{G_n-i}}, aP)$
and $\mathrm{ME_i}$ believes $\mathrm{MME_j}$ believes
$(\mathrm{ID_{MME_j}}, \mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{MME_j}}, \mathrm{R_{G_n-i}}, aP)$,
then $\mathrm{ME_i}$ believes
$(\mathrm{ID_{MME_j}}, \mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{MME_j}}, \mathrm{R_{G_n-i}}, aP)$.

In steps *m) - o)*, $\mathrm{ME_i}$ verifies message from $\mathrm{MME_j}$ by using $\mathrm{SK_{ME_i-MME_j}}$ and believes that the message is from $\mathrm{MME_j}$.

$\mathrm{ME_i}$ uses aP in a message to compute abP. $\mathrm{ME_i}$ now can compute a shared session key $\mathrm{SSK_{MME_j-ME_i}}$ between $\mathrm{ME_i}$ and $\mathrm{MME_j}$.

5.2 Authentication Proof for the Remaining MEs

We need to prove that the $\mathrm{MME_j}$ which has believed $\mathrm{ME_i}$'s long-term public key in GDL uses the key to compute a long-term secret key $(\mathrm{SK_{ME_i-MME_j}})$ between $\mathrm{ME_i}$ and $\mathrm{MME_j}$. $\mathrm{MME_j}$ uses $\mathrm{SK_{ME_i-MME_j}}$ to verify $\mathrm{ME_i}$'s message. It then can believe $\mathrm{ME_i}$'s session public key, bP. Further, the proof is that $\mathrm{ME_i}$ can believe $\mathrm{MME_j}$'s session public key, aP. Both $\mathrm{MME_j}$ and $\mathrm{ME_i}$ can use aP and bP to compute a shared session key, abP. To analyze this protocol, the following assumptions are made.

(1) $\mathrm{ME_i}$ believes $\mathrm{MME_j} \overset{\mathrm{SK_{ME_i-MME_j}}}{\longleftrightarrow} \mathrm{ME_i}$.
(2) $\mathrm{MME_j}$ believes fresh $(\mathrm{TS_{G_n-i}})$.
(3) $\mathrm{ME_i}$ believes fresh $(\mathrm{TS'_{MME_j}})$.
(4) $\mathrm{MME_j}$ believes $\mathrm{ME_i}$ controls $(\mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP)$.
(5) $\mathrm{ME_i}$ believes $\mathrm{MME_j}$ controls $(\mathrm{ID_{MME_j}}, \mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{MME_j}}, \mathrm{R_{G_n-i}}, aP)$.

The steps of the proof are as follows:

a) $\mathrm{MME_j}$ believes $\mathrm{ME_i} \overset{\mathrm{SK_{ME_i-MME_j}}}{\longleftrightarrow} \mathrm{MME_j}$
and $\mathrm{MME_j}$ sees $< \mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{TS_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP >_{\mathrm{SK_{ME_i-MME_j}}}$,
then $\mathrm{MME_j}$ believes $\mathrm{ME_i}$ said
$(\mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{TS_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP)$.

b) $\mathrm{MME_j}$ believes fresh $(\mathrm{TS_{G_n-i}})$ **and** $\mathrm{MME_j}$ believes $\mathrm{ME_i}$ said $(\mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{TS_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP)$,
then $\mathrm{MME_j}$ believes $\mathrm{ME_i}$ believes
$(\mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{TS_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP)$.

The conjunction can be broken and the result is $\mathrm{MME_j}$ believes $\mathrm{ME_i}$ believes $(\mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP)$.

c) $\mathrm{MME_j}$ believes $\mathrm{ME_i}$ controls
$(\mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}), bP)$
and $\mathrm{MME_j}$ believe $\mathrm{ME_i}$ believes
$(\mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP)$,
then $\mathrm{MME_j}$ believes
$(\mathrm{ID_{G_n}}, \mathrm{TID_{ME_i}}, \mathrm{R_{G_n-i}}, \mathrm{HFS_k}, \mathrm{LAI_{ME_i}}, bP)$.

In steps *a) – c)*, $\mathrm{MME_j}$ verifies message from $\mathrm{ME_i}$ by using $\mathrm{SK_{ME_i-MME_j}}$.

After that, $\mathrm{MME_j}$ selects random number a and computes aP. It then uses bP in $\mathrm{ME_i}$'s message to compute abP. $\mathrm{MME_j}$ now can compute a shared

session key $SSK_{MME_j-ME_i}$ between MME_j and ME_i. MME_j then sends the authentication message MAC_{MME_j} $(ID_{MME_j}, ID_{G_n}, TID_{ME_i}, R_{MME_j}, R_{G_n-i}, TS'_{MME_j}, aP)$ to ME_i.

d) ME_i believes $MME_j \overset{SK_{ME_i-MME_j}}{\longleftrightarrow} ME_i$ **and** ME_i sees $< ID_{MME_j}, ID_{G_n}, TID_{ME_i}, R_{MME_j}, R_{G_n-i}, TS'_{MME_j}, aP >_{SK_{ME_i-MME_j}}$,
then ME_i believes MME_j said $(ID_{MME_j}, ID_{G_n}, TID_{ME_i}, R_{MME_j}, R_{G_n-i}, TS'_{MME_j}, aP)$.

e) ME_i believes fresh (TS'_{MME_j}) **and** ME_i believes MME_j said $(ID_{MME_j}, ID_{G_n}, TID_{ME_i}, R_{MME_j}, R_{G_n-i}, TS'_{MME_j}, aP)$,
then ME_i believes MME_j believes $(ID_{MME_j}, ID_{G_n}, TID_{ME_i}, R_{MME_j}, R_{G_n-i}, TS'_{MME_j}, aP)$.

The conjunction can be broken and the result is ME_i believes MME_j believes $(ID_{MME_j}, ID_{G_n}, TID_{ME_i}, R_{MME_j}, R_{G_n-i}, aP)$.

f) ME_i believes MME_j controls $(ID_{MME_j}, ID_{G_n}, TID_{ME_i}, R_{MME_j}, R_{G_n-i}, aP)$
and ME_i believes MME_j believes $(ID_{MME_j}, ID_{G_n}, TID_{ME_i}, R_{MME_j}, R_{G_n-i}, aP)$,
then ME_i believes $(ID_{MME_j}, ID_{G_n}, TID_{ME_i}, R_{MME_j}, R_{G_n-i}, aP)$.

In steps *d) - f)*, ME_i verifies message from MME_j by using $SK_{ME_i-MME_j}$ and believes that the message is from MME_j.

ME_i uses aP in a message to compute abP. ME_i now can compute a shared session key $SSK_{MME_j-ME_i}$ between ME_i and MME_j.

6 Security Analysis

In this section, we have analyzed the security of SE-GA as follows.

6.1 Entity Mutual Authentication:

The main goal is to have an authentication between MME and ME in order to create a secure channel for sending data. For the first ME, it will authenticate itself with the home facilitator server (HFS) because the information of ME and the group is at the Home of ME. After ME has confirmed its success, the Home will send ME's group detail list (GDL) to MME. MME trusts ME and the authentication message from ME because MME gets a correct response from ME's Home.

The rest of the group members can authenticate directly with MME because the information of MEs and the group has been sent to MME after the first ME has finished its authentication process.

For example, ME and HFS have a shared key (SK_{ME-HFS}) generated from Diffie-Hellman key exchange in the initialization stage. For authentication of the first ME, ME generates $AUTH_i$ and sends it to the MME. The MME verifies Home of ME from $AUTH_i$ and then forwards $AUTH_i$ to the Home. Home verifies the first ME by function MAC_i which is computed by using a shared key (SK_{ME-HFS}) between ME and Home. For authentication between ME and MME, MME uses the information obtained from ME's Home to generate a key (SK_{MME-ME}) between ME and MME to validate MAC_q. If it is valid, MME trusts ME and sends $AUTH_{MME_j}$ to ME. ME checks the MME by verifying MAC_{MME_j} in $AUTH_{MME_j}$ using the key (SK_{MME-ME}) between ME and MME. If the verification passes, ME believes MME.

For the rest of the group, the mutual authentication between ME and MME is made by using function MAC_q and MAC_{MME_j} which are computed by using a long-term secret key (SK_{MME-ME}).

6.2 Confidentiality

After the authentication process, the key data used for generating the session key (SSK/KGK) between MME and ME is abP computed by using the Diffie-Hellman key exchange. The session key (SSK/KGK) is utilized to encrypt data between ME and MME. Thus, SSK/KGK can provide the data confidentiality.

6.3 Data Integrity

The integrity of messages between ME and MME, and between ME and Home are controlled by MAC function calculated from key SK_{MME-ME}, SK_{HFS-ME}, respectively. These keys are computed by using the Diffie-Hellman key exchange and known only between the two parties. Then every message sent in the protocol has a MAC function to achieve integrity control.

6.4 Enhanced Privacy-Preservation

For the first time when ME registers with the HFS, the ME gets a pair of permanent/temporary identity (PID_{ME}/TID_{ME}) to register in 3GPP networks. In the real case, ME does not send PID_{ME} into the communication network without protection because PID_{ME} is ME's privacy which may cause harm if it is sniffed. In SE-GA protocol, ME can send TID_{ME} into the communication network to the other party with MAC and the party can verify TID_{ME} by MAC function. In addition, in the case that the network needs ME to send PID_{ME} to the home network, the PID_{ME} may be encrypted with the long-term secret key between ME and HFS.

6.5 Secure Key Derivation

In the SE-GA protocol, SSK_{MME-ME} is created from a function which uses a shared secret between MME and

ME. As described in Section 5, MME and ME send a session public key of their own (aP/bP) to compute a shared secret abP between them. This abP is computed by making use of Diffie-Hellman key exchange which is secure. After that, both MME and ME use abP to generate $\mathrm{SSK}_{\mathrm{MME-ME}}$.

6.6 Key Forward/Backward Secrecy (KFS/KBS)

In the SE-GA protocol, the session public keys (aP/bP), which are used to compute session key, are sent between MME and ME, while the long-term secret $\mathrm{SK}_{\mathrm{ME-MME}}$ is calculated from a long-term public/private keys of MME and ME respectively. Then, the session public keys are not related to the calculation of the $\mathrm{SK}_{\mathrm{ME-MME}}$. In addition, the SSK key value between ME and MME is very difficult to attack. Because this value is based on abP and known only between ME and MME, then the KFS/KBS can be achieved.

6.7 Group Key Forward/Backward Secrecy (GKFS/GKBS)

When group members join or leave the group, the group key needs to update in order to preserve backward and forward secrecy. Up to now, several protocols have been proposed for dynamic group key agreement, such as Pipat [5] and Zhu [21]. After updating the group key, the group will send a group's information such as the public keys of new members/leaving members, group members' numbers to each member's Home. Then the member who has joined or left cannot know any information before joining or after leaving.

6.8 Resistance to Replay Attack

While MME and ME are communicating, authentication messages are sent with timestamps and random numbers, thus preventing replay attacks. For example of case 1, between MME and ME, there is a chance of replay attack, so while ME sending a message to MME in Step 1 to request services, a timestamp ($\mathrm{TS}_{\mathrm{G_n-i}}$) is included into the message. Similarly, when MME responds to ME in Step 4, a timestamp ($\mathrm{TS}'_{\mathrm{MME}}$) is attached to the message to prevent replay attack.

6.9 Resistance to Redirection Attack

Because the authentication message ($\mathrm{AUTH_i}$) from ME included with $\mathrm{LAI_{ME}}, \mathrm{MAC_q}$ and $\mathrm{MAC_i}$. The $\mathrm{LAI_{ME}}$ indicates the BS which ME contacts at that time. If the MME forwards $\mathrm{AUTH_i}$ to HFS, then the HFS uses $\mathrm{LAI_{MME_j}}$ to compare with $\mathrm{LAI_{ME}}$. In the case $\mathrm{LAI_{ME}} = \mathrm{LAI_{MME_j}}$, the HFS computes $\mathrm{MAC'_i}$ and compares with $\mathrm{MAC_i}$ in Step (3) of authentication for the first ME. If $\mathrm{MAC'_i} = \mathrm{MAC_i}$ then HFS accepts the authentication. It rejects the authentication if the verification of $\mathrm{MAC_i}$ fails. For the remaining ME, the MME uses $\mathrm{LAI_{MME_j}}$ getting from the BS to compare with $\mathrm{LAI_{ME}}$ embedded in $\mathrm{AUTH_i}$. If $\mathrm{LAI_{ME}}$ has the same value as $\mathrm{LAI_{MME_j}}$ then MME verify $\mathrm{MAC_q}$ with $\mathrm{MAC'_q}$. Thus, SE-GA protocol can prevent the redirection attack.

6.10 Resistance to Man-in-the-Middle Attack

During the first confirmation of ME, an attacker may disguise as MME to sniff the information. Then the attacker disguises as the ME and sends the information to the real MME. As the attacker does not know the value b, he/she may try to perform man-in-the-middle attack by replacing bP with b_1P. However, it cannot fool the Home because the attacker does not know the secret key $\mathrm{SK}_{\mathrm{ME-HFS}}$ which is utilized to compute MAC between ME and its Home.

In the case of the remaining ME, the secret key ($\mathrm{SK}_{\mathrm{ME-MME}}$) is utilized to protect messages between ME and MME. If an attacker changes messages, the MME can know messages which are not sent from the real ME. Thus, the protocol can prevent a man-in-the-middle attack.

6.11 Resistance to DoS Attack

While performing the authentication process, a malicious ME can run DoS attack either on HFS or MME. If a malicious ME forges the message, HFS or MME can detect the forged message by checking TS and comparing LAI in the message from the ME with LAI from MME.

6.12 Resistance to Impersonate Attack

The SE-GA protocol makes use of each ME's long-term private and public keys to achieve secure authentication between ME and MME. It is very difficult for an ME to disguise itself as another ME.

Table 6 shows the comparison of security and flexibility based on an actual usage in some group authentication protocols. By the comparison, we see that SE-GA is better than other protocols.

AK: Authentication key; RMA: resistance to man-in-the-middle-attack; RRA: resistance to redirection attack; GMD: group members can come from the different home networks; GMS: Group members can use different networks simultaneously; GDO: group members disguised as others.

* The first ME uses a pre-shared key which it got from the Home in the initial stage to authenticate with the Home in order to use the network service, while the remaining MEs use mainly the group key to authenticate with the MME.

Table 6: Comparisons of the proposed protocol with some schemes

Protocol / Features	SE-AKA	GLARM	SE-GA
AK	Symmetric Keys & Group Key*	Symmetric Keys**	Diffie - Hellman***
RMA	Yes	Yes	Yes
RRA	Yes	Yes	Yes
GMD	No	No	Yes
GMS	No	No	Yes
GDO	Yes	No	No

** Each ME uses the symmetric key defined by its Home when it first registered with the Home in order to authenticate itself with the service network.

*** The key used in the authentication process can be created on the fly between the two parties by making use of the Diffie-Hellman key exchange.

7 Conclusions

In this work, we have developed the SE-GA protocol that assists group authentication on LTE networks. The authentication protocol uses the long-term private keys and public keys between parties to create shared secret keys used in the authentication process. By using this technique, SE-GA can be flexible and scalable. It helps the group members to be able to work simultaneously on different LTE networks. In addition, group members can be from different Homes. In the protocol, the authentication process is divided into two steps, the authentication of the first machine which tries to connect to a service network and that of the remaining machines. The first machine needs to authenticate itself with its Home, while the remaining machines can authenticate with the service MME. This reduces the providers' network traffic as well as network delays.

During the initialization of SE-GA, the network will be a little crowded because each group member has to send group information to its Home. However, during the authentication of the group members excluding the first one, SE-GA needs only three steps for the authentication of each member while the former SE-AKA needs at least four steps.

In this paper, we provided an authentication proof by using the well-known BAN logic. Security analysis of the proposed protocol is also given and a comparison of our protocol with SE-AKA and GLARM was demonstrated. According to the comparison, we can see that the proposed protocol outperforms the former ones.

References

[1] S. B. Babu and P. Venkataram, "A dynamic authentication scheme for mobile transactions," *International Journal of Network Security*, vol. 8, no. 1, pp. 59–74, 2009.

[2] M. Burrows, M. Abadi, and R. Need ham, "A logic of authentication," *ACM Transactions on Computer Systems*, vol. 8, no. 1, pp. 18–36, 1990.

[3] J. Cao, M. Ma, and H. Li, "A group-based authentication and key agreement for mtc in lte networks," *IEEE Globecom (Globecom'12)*, pp. 1017–1022, 2012.

[4] Y. W. Chen, J. T. Wang, K. H. Chi, and C.-C. Tseng, "Group-based authentication and key agreement," *Wireless Pers Commun*, vol. 62, no. 4, pp. 965–979, 2012.

[5] P. Hiranvanichakorn, "Provably authenticated group key agreement based on braid groups - The dynamic case," *International Journal of Network Security*, vol. 19, no. 4, pp. 517–527, 2017.

[6] M. S. Hwang, S. K. Chong, and H. H. Ou, "On the security of an enhanced umts authentication and key agreement protocol," *European Transactions on Telecommunications*, vol. 22, no. 3, pp. 99–112, 2011.

[7] M. S. Hwang, C. C. Lee, and W. P. Yang, "An improvement of mobile users authentication in the integration environments," *International Journal of Electronics and Communications*, vol. 56, no. 5, pp. 293–297, 2002.

[8] C. Lai, H. Li, R. Lu, and X. Shen, "Se-aka: A secure and efficient group authentication and key agreement protocol for lte networks," *Computer Networks*, vol. 57, pp. 3492–3510, 2013.

[9] C. Lai, R. Lu, D. Zheng, H. Li, and X. Shen, "Glarm: Group-based light weight authentication scheme for resource-constrained machine to machine communications," *Computer Networks*, vol. 99, pp. 66–81, 2016.

[10] C. C. Lee, M. S. Hwang, and L. H. Li, "A new key authentication scheme based on discrete logarithms," *Applied Mathematics and Computation*, vol. 139, no. 2, pp. 343–349, 2003.

[11] C. C. Lee, M. S. Hwang, and I. E. Liao, "A new authentication protocol based on pointer forwarding for mobile communications," *Wireless Communications and Mobile Computing*, vol. 8, no. 5, pp. 661–672, 2008.

[12] C. C. Lee, M. S. Hwang, and W. P. Yang, "Extension of authentication protocol for gsm," *IEEE Proceedings – Communications*, vol. 150, no. 2, pp. 91–95, 2003.

[13] C. C. Lee, I. E. Liao, and M. S. Hwang, "An efficient authentication protocol for mobile communications," *Telecommunication Systems*, vol. 46, no. 1, pp. 31–41, 2011.

[14] Y. Liu, C. Cheng, J. Cao, and T. Jiang, "An improved authenticated group key transfer protocol based on secret sharing," *IEEE Transactions on Computers*, vol. 62, no. 11, pp. 2335–2336, 2013.

[15] Y. Liu, C. Cheng, T. Gu, T. Jiang, and X. Li, "A lightweight authenticated communication scheme for smart grid," *IEEE Sensors Journal*, vol. 16, no. 3, pp. 836–842, 2016.

[16] H. H. Ou, M. S. Hwang, and J. K. Jan, "A cocktail protocol with the authentication and key agreement on the umts," *Journal of Systems and Software*, vol. 83, no. 2, pp. 316–325, 2010.

[17] H. H. Ou, I. C. Lin, M. S. Hwang, and J. K. Jan, "Tkaka: Using temporary key on authentication and key agreement protocol on umts," *International Journal of Network Management*, vol. 19, no. 4, pp. 291–303, 2009.

[18] S. F. Tzeng, M. S. Hwang, "Digital signature with message recovery and its variants based on elliptic curve discrete logarithm problem", *Computer Standards & Interfaces*, vol. 26, no. 2, pp. 61–71, Mar. 2004.

[19] F. Wang, C. Chang, and Y. Chou, "Group authentication and group key distribution for ad hoc networks," *International Journal of Network Security*, vol. 17, no. 2, pp. 199–207, 2015.

[20] S. Wu, Y. Zhu, and Q. Pu, "Security analysis of a cocktail protocol with the authentication and key agreement on the umts," *IEEE Communications Letters*, vol. 14, no. 4, pp. 366–368, 2010.

[21] H. F. Zhu, "Secure chaotic maps-based group key agreement scheme with privacy preserving," *International Journal of Network Security*, vol. 18, no. 6, pp. 1001–1009, 2016.

Biography

Boriphat Kijjabuncha received the B.S. degree from Silpakorn University, Thailand, in 2000 and M.S. degree in Applied Information System at School of Applied Statistics, National Institute of Development Administration (NIDA), Thailand. He has worked as a lecturer of Information and Communication Technology Department at Silpakorn University, Thailand since 2006. His research interest is in information security.

Pipat Hiranvanichakorn received the B.E. degree in Electrical Engineering from Chulalongkorn University, Thailand, in 1977 and the M.E. and D.E. degrees in Information Processing from Tokyo Institute of Technology, Japan, in 1982 and 1985, respectively. He has worked as an associate professor of Computer Science at School of Applied Statistics, National Institute of Development Administration, Thailand until 2016. His current research interests include natural language processing, computer networks, cryptography and information security.

URLDeep: Continuous Prediction of Malicious URL with Dynamic Deep Learning in Social Networks

Putra Wanda[1,2] and Huang Jin Jie[1]

(Corresponding author: Huang Jin Jie)

School of Computer Science & Technology, Harbin University of Science and Technology[1]
Xuefu Road No.52, Harbin, Heilongjiang 150080, China
University of Respati Yogyakarta, Indonesia[2]
(Email: wpwawan@gmail.com, huangjinjie163@163.com)

(Received May 20, 2018; Revised and Accepted Aug. 18, 2018; First Online Mar. 2, 2019)

Abstract

Malicious URLs are one of the cybercrimes techniques in digital services. They spread unsolicited content and attack victims. Attacking will make them victims suffer from scams activities such as theft of money, compromise identity, install malware and so forth. In this paper, we propose URLDeep, a deep architecture by novel CNN. Instead of using conventional CNN, we use Dynamic CNN. It can assign a similar signal in the same CNN channel. URLDeep's graph is dynamically updated after each layer of the network. Demonstrated by the experiments, the results of classification accuracy have produced useful accuracy. URLDeep is a novel deep learning framework to learn a nonlinear URL address.

Keywords: Dynamic CNN; Malicious URL; Prediction; Social Networks

1 Introduction

Social networks application have grown tremendously. It has a significant impact on the youth generation lifestyle [1]. It has built a digital environment become part of activities in our daily life. Conventional security technique like cryptography is no longer suitable for a dynamic environment or even common authentication [3,5,18]. So, dynamic environment in social networks such as peer to peer social networks [19] needs efficient security technique.

The machine learning approaches have shown successful performance, particularly when implementing a new concept called Convolutional Neural Networks (CNN) [9] is one of Deep learning architecture and are becoming increasingly popular in the current research. CNN's produced a promising performance for text classification [21]. CNN is different from other neural networks because of convolutional layer concept. Output feature in CNN convolves the previous feature maps with a set of filters in Deep Network. In this study, we designated a novel model of CNN's to learn Malicious URL feature for detecting suspicious address. However, instead of using conventional CNN which uses a static layer in the convolution process, we propose dynamic layer computation in convolutional process.

URLDeep, the underlying deep neural network is a Dynamic Convolutional Neural Network (CNN). In URLDeep, continuous graph updates beneficial to recompute the graph using the k-max concept. Therefore, it can better deal with the problem of data noise, alignment, and other data variations. Demonstrated by the experiments, the results of detection accuracy have shown that URLDeep algorithm is handy for Continuous prediction for malicious URL.

We will summarize the main contributions of our work as follows:

1) We present a novel continuous prediction method of malicious URL, particularly in social networks with Dynamic Deep Learning concept. It used to identify feature representation and learned in useful CNN algorithm.

2) We demonstrate URLDeep with the new dynamic graph in CNN. It is different with conventional CNN operations and has produced better accuracy.

3) With benchmark datasets, we show complete analysis and testing of k-max pooling function and attain state-of-the-art performance.

URLDeep is a novel framework to extract and learn URL address. In this method, we implement Dynamic Convolutional Neural Networks to both characters and words of the URL. It is to learn the URL embedding in the framework. The approach can capture more types of semantic information. As long as we know, it was not

possible by the existing models. The model learns a representation of the raw URL string directly without any designed help from the expert.

2 Related Work

The first alphabet of each word in the title of each section must be Capital Letter.

2.1 URL Detection with Machine Learning

A model developed to detect suspicious URL filtering based on reputation users in the networks [2]. Fast feature extraction technique [17], an approach with designed Multi-layer perception [10], and in big data security area, a study presented a model to secure big data with data mining analysis [15]. Behavioral analysis of attackers [8], prediction based on Community Detection [11].

Many researchers have used the Neural Networks algorithm for addressing issues in a computer environment. In a malicious URL case, many research proposed Machine Learning. A study discussed different machine learning techniques and unsupervised learning. It may able to detecting malicious URLs with the semi-supervised system to detect malicious URL or using Active Learning for Malicious URL Detection with Weighted Text-Based Features [16].

Machine learning is widely used for various cases because understand for non-linear relationships and own robust to outliers. However, it still owns weakness for unconstrained data and prone to overfitting in the training process. A widely used algorithm in machine learning such as SVM suffers over-fitting problem from optimizing the parameters to model selection. In SVM kernel models are very sensitive to over-fitting selection criterion in training process. Therefore, the researcher developed a new advanced algorithm like deep learning.

2.2 URL Detection with Deep Learning

Deep learning is a part of machine learning that is a set of algorithms imitate the structure and function of the brain. It operates on raw input signals and automating the process of feature extraction Deep learning is becoming increasingly popular in solving various applications, one of them is the authentication process.

Deep learning, a subfield of machine learning A study proposed, the eXpose neural network, which uses deep learning with convolutional neural network approach, is to detect artifacts like potentially malicious URLs, file paths, named pipes, and registry keys. The model will learn to simultaneously extract features and classify using character-level embedding [7].

Newly study in malicious URL with deep learning, called URLNet, the paper proposed an end-to-end deep learning framework to learn a nonlinear URL embedding for Malicious URL Detection directly from the URL. Mainly, the model implemented conventional CNN to both characters and words of the URL [6].

A recent study securing URL address, a technique using Event De-noising Convolutional Neural Network for Sequence Detection in malicious URL. The paper proposed a system for detecting malicious URL sequences from proxy logs with a low false positive rate [13].ED-CNN is a particular CNN to reduce the adverse effect of benign URLs redirected from compromised websites such as malicious URL sequences.

In this model, instead of using conventional CNN in deep learning as mentioned above, we propose URLDeep, Deep Learning approach based on Dynamic Convolutional Neural Network (D-CNN). The study uses Dynamic CNN model for detecting malicious URL. Different from conventional CNN, URLDeep can assign similar signal parts to the same CNN channel. In the network, the k-max pooling of a point changes from layer to layer.

3 Model

3.1 Conventional CNN

In common CNN model, the filters are the only parameters of the convolutional layer during training. In CNN algorithm, tensors are essential. In CNN, beginning with the input, intermediate representation, and parameters in computing process are all tensors.

The conventional CNN many identical neurons among the layers. In the CNN model, a computational process run large models computation with a little number of parameters. In Conventional Convolution Neural Network, while the layer receives a single input (the feature maps), it will compute feature maps as its output by convolving filters across the feature maps. The parameters of the convolution layer called filters and back-propagation model used to learn during training.

3.2 URLDeep Neural Network

In this section, we describe the process of URLDeep based on dynamic CNN, which is a generalization of the convolution layer. The discussion explores how the two layers differ from one another regarding input, output, the forward pass and backward pass.

URLDeep, a novel framework with dynamic max pooling in CNN concept. URLDeep model can re-compute the graph using determined layer number, assign similar signal parts to the same CNN channel, and it will able to determining the optimal alignment of weights. Therefore, it can better deal with the problem of data noise in URL address and other data variations. In this model, URLDeep implements dynamic max pooling in computing CNN network.

In contrast to the conventional CNN layer, the URLDeep layer receives two inputs within the operation.

The first input is the previous layer of the features maps and the second is the filters. In this study, URLDeep receives a URL string as input and applies dynamic CNNs to both characters and words in computing the URL address.

Character-level CNN will identify unique characters in the dataset and transform each character into a numerical vector. In this phase, a URL sequence is converted to a matrix representation; Then, the convolution process will compute the matrices. The part will identify critical information which contains maliciousness. Word-level CNN will identify unique words in the training dataset with special characters in URL address. The process will get a matrix representation of the URL or sequence of words. Word-level phase processes to identify useful patterns from certain words that appearing together.

Forward Pass:

$$y_j^n = \sum_i k_{ij}^n * x_i^n. \tag{1}$$

In this process, the first network computes the features maps as the input layer to the dynamic CNN. In the Equation (1), $x_i^n$is the i-th input feature map of the sample n and k_{ij}^n is the ij input kernel of the sample n.

Backward Pass:

$$\begin{aligned} \frac{\partial l}{\partial x_i^n} &= \sum_j \left(\frac{\partial l}{\partial y_j^n}\right) * \left(k_{ij}^n\right) \\ \frac{\partial l}{\partial k_{ij}^n} &= \left(\frac{\partial l}{\partial k_j^n}\right) * x_i^n. \end{aligned} \tag{2}$$

The function of partial derivative computes the gradient of l with respect x_i^n. The values of the gradient calculated by the partial derivative function $\frac{\partial l}{\partial x_i^n}$ and passed to the first layer of the network. Partial derivative in Equation (2) computes the l with respect to k_{ij}^n .

In Dynamic CNN, when one model trained, the dynamic assignment approach is adapted to fit the input signal on a segment basis. The dynamic assignment contains two steps; they are Data Partition and Channel Fitting.

Data Partition process, each activity class, owns the trained model. Then, it will forward it into N parts which correspond to the N channels in the D-CNN. The features partitioned into N parts as well.

Channel Fitting, features which are similar to the same model part go to the same channel in the D-CNN. The distance d between the partitioned features and the model part depicts as Equation (3).

$$d = \frac{1}{n}\sum_{t=1}^{n} d_t, \tag{3}$$

where n is the size of the partitioned feature and t is acceleration signal at the time.

The N model parts correspond to N channels of CNN. In this model, the channel referred to as one path containing convolution and pooling operations in the CNN. Then, the results of N channels are concatenated to build the feature map. Last, two feature maps of gravity and body combined as input to another convolution and pooling operations to capture the correlation between two features.

4 Our Approach

4.1 General Idea

Nowadays, Deep Learning research is growing tremendously. It used to address the various problem in human life such as Image, Video, Audio, Traffic Management, Internet of Things until Biological Data Processing [4, 12, 20].

The various study has been done for detecting malicious URLs. The most common approach is blacklist filtering. The technique is simple but not scalable, though some enhanced approaches were utilizing fuzzy matching. Other study tried to use machine learning (ML) to extract features from URL strings. Recently, an approach using deep learning (DL) to extract features automatically. It applied a mechanical approach to generate feature vectors from URL strings [14].

Instead of using Common Deep Learning model, the model in this research has proposed a novel framework with dynamic CNN concept; it is to predict malicious URL in Social Networks in real-time. It is to classify a new URL as malicious or benign URL. To implement the concept, we will formulate and compute the problem as a binary classification task. In the research, consider S that represents URL is a set of URL$\{(u_1, y_1) \cdots (u_s, y_s)\}$, where u_1 for $s = 1$. $y_s \in \{-1, +1\}$ denotes URL's label. Malicious URL is $y = +1$ and $y_s = -1$ is the benign URL.

The first step in classification procedure is to get feature representation $u_s \rightarrow x_s$ where $x_s \in R^n$ is the n-dimensional feature for u_s as vector represenation of URL. Then, the second step of the model computes prediction CNN function $f : R^n \rightarrow R$. R is the parameter for score prediction of the URL x. The formula $Y_t = sign(f(x_s))$ is to minimize the mistake amount in data training.

This study proposed a novel method that can reach accurate results with dynamic graph concept. However, it will heighten robustness in detecting risk when compared with conventional detecting approaches. This study uses a dataset consisting of million URL address. We design distinctive classifiers for each modality for the mobile profiling and activity behavior. Notably, the URLDeep model runs the process by obtaining a URL string as input, then uses dynamic CNN concept for computing the URL.

This research, re-compute the graph using dynamic pooling in k-max with nearest neighbors. It will be pos-

sible and beneficial for getting efficient to make a continuous prediction in malicious URL. The model is very different from conventional CNN which is using a fixed input in the layer. It calculates effective operation in k-max pooling along the CNN networks and is a crucial distinction of conventional CNN. This study adopts an unsupervised learning model that detect suspicious URL by merely analyzing unlabeled data with the numerical vector. The unsupervised learning has no teacher; it used to malicious detection in social networks.

4.2 Lexical Feature

In training cases, a raw URL converted to a compatible feature vector $u \to x$ before starting training a prediction model. The machine learning uses to calculate the training process. To achieve an effective result, we choose the feature representation to classify different of the features of the URLs.

In this framework, we will focus on the lexical features approach. It is to obtain the feature representation for the URLs. Lexical features are the technique to obtain feature based on the properties of the URL name or string.

The Lexical features technique is very famous and useful to obtain features information of the URL by calculating string of the URL address. Deep learning is a new technique for training and testing process in various cases include in URL detection cases. This model adopts Lexical features for the first stage to convert raw URL u become a feature vector x. In this process, a URL split into words and characters. The model will identify characters and words by transforming each word w_i becomes a feature. Particular of M feature, u_s will be mapped to a vector $x_s \in R^M$.

4.3 URLDeep Architecture

URLDeep, a Dynamic CNN framework for detecting Malicious URL real-time. This model will compute the dynamic CNN networks with a vector representation of the URLs. CNNs have achieved extraordinary success in various tasks. It can learn the salient features from URL string value automatically.

The URLDeep framework based on deep neural network concept, this is a novel framework with dynamic CNN approach for detecting malicious URLs. It is used to learn structural information about the URL. Particularly for URL based on at both the character-level and word-level. This part will describe the malicious URL detection by dynamic CNN model. The process uses Dynamic CNN for computing and classifying malicious or benign URL by computing matrix representation $u \to x \in R^{L \times m}$.

In the model, a URL sequence u is combinatoin of words or characters and separated by special characters. Firstly, we have to obtain URL's matrix representation $u \to x \in R^{L \times m}$ while instances x consist of neighbors components $x_i, i = 1, \cdots, L$ in the sequence. Componens comprise a word or a character of the URL. Each of component is calculated by $x \in R^m$, is an m-dimensional vector. In this study, firstly we initialize the embedding matrix randomly, and use end-to-end optimization to learn the matrix. The m-dimensional representation is an embedding vector which is produced by an embedding matrix.

Instead of using static max-pooling and fixed graph concept (used in conventional CNN). In this study, we propose dynamic pooling and activation output operation. The concept of dynamic k-max pooling can assign similar signal parts to the same CNN channel. It is a novel approach to achieve authentication prediction in malicious URL issues. it can better deal with the problem of data noise, alignment, and other data variations

In the URLDeep, k max pooling k is a function of the input length and network's depth with the equation:

$$k_l = max(k_{top}[\frac{C-c}{C}V_{cw}]). \quad (4)$$

In the Equation (4), c is several of new convolution layer, and C represents the amount of convolution layer, k_{top} is the pooling parameter with the fixed value, it is the top of the convolutional layer. Then to detect URLs whether the malicious or benign URLs, the function need V as vector value of a character or word URL features.

It is to enhance the accuracy of multi-classifier systems and better deal with the problem of data noise, ill alignment, and other data variations. Growing URL addresses in Social Networks environment need accuracy in computing, it is to yield best real-time in the prediction process. Figure 1 describes the process of continuous prediction for malicious URL detection with URLDeep model. It implements Dynamic CNN concept for building algorithm.

This model has a two-tier of calculating matrix in URL detection. It consists of Character level and Word level. The first tier of this architecture called Character level URLDeep, it is to mapping URL address based on character. This phase will convert any characters in the address become unique values. Representation of URL address in numeric values can produce a particular matrix. The phase produces matrix representations result. URLDeep applies the concept of dynamic graph computation by calculating dynamic k-max pooling and output activation map along layers.

The second tier called Word level URLDeep. It is by mapping various URL address based on the word. In this phase, it converts any URLs address in the address become unique values which own different values among the words. Value of word will produce matrix representation; it will ease the calculation of CNN layer. This process also implements the concept of dynamic graph computation with dynamic k-max pooling and output activation map value o.

The next phase of the URLDeep architecture is processed to get a final score of classification. After calculating of matrix representation process, the architecture of the URLDeep shifts to classification. The process called a

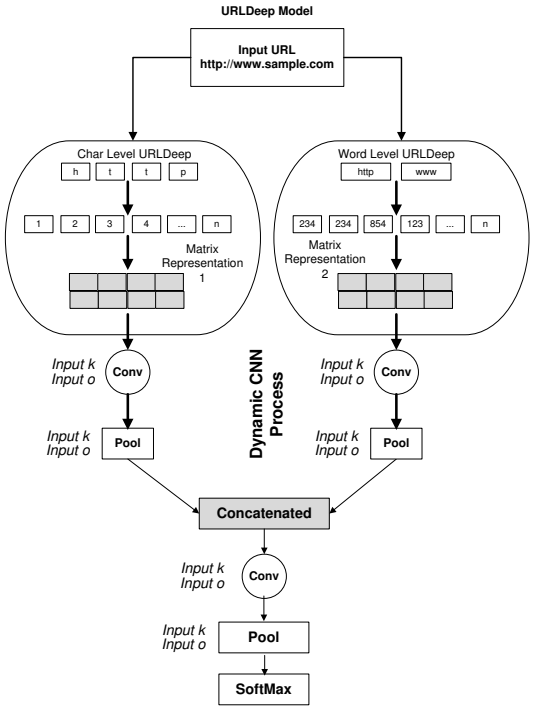


Figure 1: The URLDeep Model with deep learning framework to learn a nonlinear URL address with Characters and Word level. Calculating of matrix representation is to detect Malicious URL directly from the vectors with dynamic k-max pooling k and o is a number of output activation map. It is to learn the URL embedding in the framework

fully connected layer (FC). FC will compute that outputs a vector of K (the number of classes) dimensions in the classification process. The vector owns the probabilities for each class. In the top layer, we use a designated SoftMax function for calculating loss function of the networks to achieve an accurate result.

Character Level URLDeep: This phase presents the main ideas for building Character level URLDeep for Malicious URL Address Detection. It will learn to embed properties about the URL sequence characters. Firstly, in the dataset corpus, We identify all the unique alphanumeric and special characters. To calculate the URL representation into vector value, We set the length of the URL sequence $L_1 = 180$ characters. The phase truncates the characters which the length more than 180.

In the URLDeep process, the character will be embedded into a m ?dimensional vector. In this model, the value of m is dynamic for characters between 16 to 32. During the training process, the embedding value initialized and learned randomly. With the embedding concept, each URL u converts a character into a matrix $u \rightarrow x \in R^{L \times m}$. The formula computes dynamic m value and length of the URL address $L_1 = 180$. This is novel approach for calculating matrix representation in URL address.

Character-level URLDeep can obtain an embedding for new URLs easily in the dataset. The phase changes all the URLs become the URL matrix $x_t \forall t \in R$ as the training data. In this process, it uses dynamic Max-Pooling and followed by fully connected layer. The pooling result will be concatenated with other part of URLDeep.

Word Level URLDeep: In this phase, we identify the unique words in URLs dataset. Unique words depend on the size of data training. In each URL, new words can appear and is different with Character URLDeep with small unique character and fixed value. In this model, we identify unique words with Lexical Feature approachment. All of the unique words are a sequence of alphanumeric characters (including dish character '-' or '_'), length of the URLs number in the model is $L_2 = 180$. All of the unique words make a dictionary for training dataset.

In the next phase, the model gets m-dimensional vector representation. In this study, we use dynamic m value between 16 to 32, that is mean each word comprised to a 16-dimensional vector or 32-dimensional vector. For W unique words, we need to compute a matrix $W \in R^{W \times k}$. When computing with CNN algorithm, the representation of URLs are transformed to matrix representation $L_2 xk$. Word Level URLDeep uses the identic CNN model with Character Level URLDeep.

The last URL matrix representation is the sum of calculation of the above matrices included Word Level URLDeep URL_w with Character Level URLDeep URL_c.

5 Experiment Result

5.1 Large Scale Dataset

In this research, we collected an extensive database of labeled URLs from VirusTotal and PhishTank. VirusTotal service used to validate a URL whether it is malicious or benign. We also collected about 30,000 malicious URLs from PhishTank.

We used a set of benign and malicious URLs for training and testing process. Then we create an annotated database to train and test the URLs. Table 1 depicts URL Dataset corpus for the research.

Table 1 depicts URLs Dataset from VirusTotal and PhishTank.

Table 1: Dataset testing in URLDeep model

	Benign	**Malicious**	**Total**
Training	4,983,425	1.016,575	6,000,000
Testing	9,066,850	933,150	10,000,000
Total	14,050,275	1.049,725	16,000,000

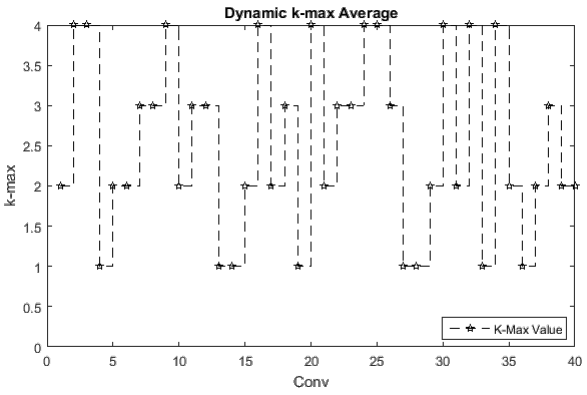


Figure 2: Value of k-max average to compute in the CNN's layer. Dynamic value of activation map will be computed with k value in CNN layer

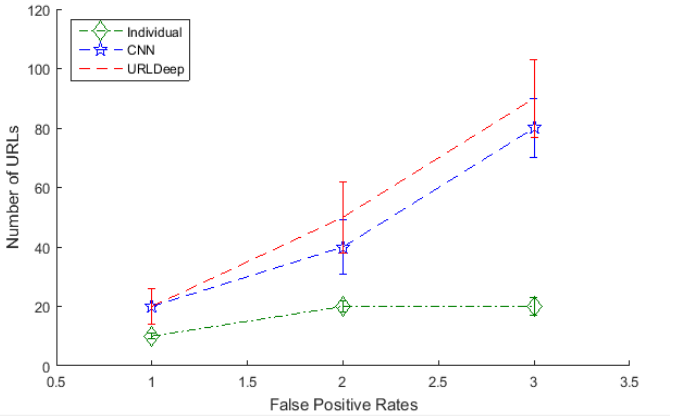


Figure 3: Value of Undetection rate of URL sequences based on FPR calculation

5.2 Analysis

In this study, we compute the character level of URL and word level of URL for detecting malicious URL in a social network environment. Run several training processes for URL addresses with dynamic CNN algorithms. The detection process determines whether an URL is a benign or malicious category.

This study produces the optimal result by URLDeep model-based dynamic CNN architecture. We see dynamic CNN by k-max pooling operation resulted in good accuracy to classify whether the URL address benign or malicious URL. Besides, it is a novel approach for securing social networks when accessing new address over the network. We find a significant increase in CNN graph's performance within dynamic k-max pooling operation for training data and classify URL address.

This model implements dynamic max-pooling of k and activation map value o. Parameter k is kernel parameter used to compute max-pooling layer. Parameter o is total of activation map in the layer. The model applies random activation map o and dynamic value of k-max pooling. Figure 2 shows dynamic k-max pooling value in URLDeep architecture.

This simulation process has used average neuron numbers to train the dataset. It produces relatively high accuracy with k-max pooling operation. In our experiment, more layers and neuron numbers of CNN could not engage to improve the predictive capability. It enlarges resource in the computing process. Therefore a limitation of neuron number and k-max pooling is an effective method to achieve efficient network in dynamic.

Beside of using pooling concept in each layer of neural network, this model has applied Stochastic Gradient Descent (SGD) with Optim. We also implement local variable concept to computing the loss function. To support the local variable concept, the model uses Optim library and Gradparams. SGD based Optim able to calculate the gradient of the loss concerning the weight with various learning rate.

The model train Stochastic Gradient Descent with Optim parameter to train and calculates loss value in each layer of dynamic CNN. It achieves a better result in dynamic CNN network. Algorithm 1 describes an algorithm designed SGD with Optim parameter.

Algorithm 1: Designed of SGD with Optim
Modeling Multi-Layer Perception (MLP)
Determine Input $x?R$ and Hidden Layers HU
Choose ReLU $f(x) = \max(x, 0)$
Determine the size of dataset or batch file
Define Learning rate: LR$>$ 0, $LR?R$
Train SGD with Optim
Print Tensor

In training process, this study various transformed data of the different user become different elements called E. the above graphs show the element $E_1 \cdots E_n$ when running a detection process. Dynamic CNN will analyze data transforming when user interaction in a social network.

We evaluate detection performance results of malicious URL sequences. The URLDeep has implemented general indexes to detection performance called False Positive Rates (FPR), is malicious URL which benign URL categories.

In another hand, the study evaluated the false alerts of malicious URL detection. The URLDeep detects characteristics of malicious redirection with a low FPR. Figure 3 depicts the false alert with FPR indexes.

It is noteworthy that the Undetection rate with URLDeep was significantly high compared to CNN and individual approaches. The URLDeep calculates malicious URL detection based on exploit URLs feature extraction. Besides, CNN approach detects malicious URL sequences based on landing URLs. Based on our experiment, the individual and CNN approach detected several malicious URLs with the sparse result. In contrast, the URLDeep model successfully detected the characteristics of redirections to exploit URLs, and it has produced a

highest Undetection rate of URL sequences based on FPR calculation.

6 Conclusions

The digital environment needs real-time and adaptive security model. Common security model as cryptography is no longer suitable for securing the digital environment. Nowadays, Deep Learning becomes much more popular to overcome various issues. This paper proposes URLDeep, a Dynamic CNN for detecting malicious URLs in the social network. The URLDeep concept can assign similar signal parts to the same CNN channel. Therefore, it can better deal with the problem of data noise, alignment, and other data variations. Demonstrated by the experiments, the results of classification accuracy have produced useful accuracy. URLDeep is used to detect Malicious URL directly from the URL address features. This method can classify malicious or benign URLs in Peer-to-Peer (P2P) Social Network.

Based on the experiment, detection performance results of malicious URL sequences produced by evaluating the false alerts of malicious URL detection. The undetection rate with URLDeep model was significantly higher compared to CNN and individual approaches.

Acknowledgments

This study was supported by the China Scholarship Council under Harbin University of Science and Technology. The authors gratefully acknowledge the anonymous reviewers for their valuable comments. Thank you for all contributors in supporting the research.

References

[1] B. Akashdeep, A. Vinay, G. Sam, "Impact of social networking on Indian youth - A survey," *International Journal of Electronics and Information Engineering (IJEIE'17)*, vol. 7, no. 1, pp. 41-51, Sep. 2017.

[2] C. M. Chen, J. J. Huang, Y. H. Ou, "Efficient suspicious URL filtering based on reputation," *Journal of Information Security and Applications*, vol. 20, no. 26-36, pp. 2214-212, 2015.

[3] C. L. Cheng, Y. Chou, M. S. Hwang, "A new privacy and authentication protocol for end-to-end mobile users," *International Journal of Communication Systems*, vol. 16, pp. 799-808, 2003.

[4] G. Cheng, C. Yang, X. Yao, L. Guo and J. Han, "When deep learning meets metric learning: Remote sensing image scene classification via learning discriminative CNNs," in *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, no. 5, pp. 2811-2821, May 2018.

[5] J. Du, C. Jiang, K. C. Chen, Y. Ren and H. V. Poor, "Community-structured evolutionary game for privacy protection in social networks," in *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 3, pp. 574-589, Mar. 2018.

[6] L. Hung, P. Quang, S. Doyen, C. H. H. Steven, "URLNet: Learning a URL representation with deep learning for malicious URL detection," *Cryptography and Security*, 2018. (`https://arxiv.org/abs/1802.03162`)

[7] S. Joshua, K. Berlin, "eXpose: A character-level convolutional neural network with embeddings for detecting malicious URLs, file paths and registry keys," *Cryptography and Security*, 2017.(`https://arxiv.org/abs/1702.08568`)

[8] S. Kim, J. Kim, B. B. H. Kang," Malicious URL protection based on attackers' habitual behavioral analysis," *Computers & Security*, vol. 77, pp. 790-806, 2018.

[9] M. Kumar, H. K. Verma, G. Sikka, "A secure lightweight signature-based authentication for Cloud-IoT crowdsensing environment," *Translations on Emerging Telecommunications Technologies*, 2018. (`https://onlinelibrary.wiley.com/toc/21613915/0/0`)

[10] Z. Li-Xiong *et al.*, "Malicious URL prediction based on community detection," in *International Conference on Cyber Security of Smart Cities, Industrial Control System and Communications (SSIC'15)*, pp. 1-7, 2015.

[11] Z. Li-Xiong *et al.*, "Malicious URL prediction based on community detection," in *International Conference on Cyber Security of Smart Cities, Industrial Control System and Communications (SSIC'15)*, pp. 1-7, 2015.

[12] M. Mahmud, M. S. Kaiser, A. Hussain and S. Vassanelli, "Applications of deep learning and reinforcement learning to biological data," in *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 6, pp. 2063-2079, June 2018.

[13] T. Shibahara *et al.*, "Malicious URL sequence detection using event de-noising convolutional neural network," in *IEEE International Conference on Communications (ICC'17)*, pp. 1-7, 2017.

[14] K. Shima *et al.*, "Classification of URL bitstreams using bag of bytes," in *The 21st Conference on Innovation in Clouds, Internet and Networks and Workshops (ICIN'18)*, pp. 1-5, 2018.

[15] S. Thakur, E. Meenakshi and A. Priya, "Detection of malicious URLs in big data using RIPPER algorithm," in *The 2nd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT'17)*, pp. 1296-1301, 2017.

[16] F. Vanhoenshoven, G. Nápoles, R. Falcon, K. Vanhoof and M. Köppen, "Detecting malicious URLs using machine learning techniques," *IEEE Symposium Series on Computational Intelligence (SSCI'16)*, pp. 1-8, 2016.

[17] R. Verma, A. Das, "What's in a URL: Fast feature extraction and malicious URL detection," in *ResearchGate*, pp. 24, 2017.
[18] P. Wanda, Selo and B. S. Hantono, "Efficient message security based Hyper Elliptic Curve Cryptosystem (HECC) for mobile instant messenger," in *The 1st International Conference on Information Technology, Computer and Electrical Engineering*, pp. 245-249, 2014.
[19] P. Wanda, Selo and B. S. Hantono, "Model of secure P2P mobile instant messaging based on virtual network," in *International Conference on Information Technology Systems and Innovation (ICITSI'14)*, pp. 81-85, 2014.
[20] S. Yao *et al.*, "Deep learning for the internet of things," in *Computer*, vol. 51, no. 5, May 2018.
[21] X. Zhang, J. Zhao, and Y. LeCun, "Character-level convolutional networks for text classification," in *Advances in Neural Information Processing Systems*, pp. 649-657, 2015.

Biography

Putra Wanda. He received B.Eng in University of Respati Yogyakarta, Indonesia and M.Eng in Computer Science, Gadjah Mada University, Indonesia. Now, he is a PhD Student in Institute of Research in Information Processing Laboratory, Harbin University of Science and Technology (email: wpwawan@gmail.com).

Huang Jin Jie. He is an Associate Professor and Ph.D. Supervisorr in School of Computer Science, Harbin University of Science and Technology with speciality in Artificial Intelligence, Robotic and Control (email: huangjinjie163@163.com).

Detection of Network Protection Security Vulnerability Intrusion Based on Data Mining

Jinming Zhang

(Corresponding author: Jinming Zhang)

Department of Information Technology, Yantai Vocational College
Yantai, Shandong 264670, China
(Email: jinmingzjm@126.com)

(Received Feb. 2, 2019; Revised and Accepted Aug. 11, 2019; First Online Oct. 1, 2019)

Abstract

With the rapid development of Internet technology, network security has received more and more attention. Therefore, the detection of network protection security vulnerability intrusion has become an urgent task with some practical and guiding significance. In this paper, intrusion detection system (IDS) is taken as the research object to establish a data mining-based IDS model, the experimental results are obtained, and the relevant experimental conclusions are drawn. At the same time, it is compared with the traditional IDS, and six experiments are carried out. The output results of the detection rate, false negative rate and false positive rate of the two different methods in six experiments are obtained. The experimental conclusions that the network protection security performance of IDS using the data mining is better, and the detection capability of network vulnerability intrusion is stronger are drawn. This study provides a new route for the research on the detection of network protection security vulnerability intrusion.

Keywords: Data Mining; Intrusion Detection System; Protection Security; Vulnerability

1 Introduction

In the modern era, the network is slowly integrated into people's daily life, which has become an indispensable part of people's life. At the same time, the amount of data information on the network has also increased in abundance, which is followed by the frequent occurrence of unlawful incidents of data leakage on the Internet [3]. It not only exposes the privacy of individuals and businesses, but also poses some risks and safety hazards to individuals and businesses. Therefore, intrusion detection system (IDS) is quite important in this network environment, which can improve the security performance of network protection [10]. IDS mainly refers to a new detection mode [6] extending from traditional firewall technology, which relies mainly on intrusion detection technology [12] and monitor events in the network through corresponding working principles. At present, there are more and more researches on IDS, and methods such as data mining, statistical models, *etc.* can be used to improve and optimize them, thereby further improving the detection performance of network protection security vulnerability intrusion and the current network environment [1, 20].

In response to this problem, many experts and scholars have put forward their own opinions and views. Aparicio-Navarro *et al.* [4] believed that new and more powerful detection mechanisms need to be developed as the complexity of cyber attacks increases and proposed that next-generation IDS should be able to adjust its detection characteristics based not only on measurable network traffic, but also on available advanced information related to the protected network to improve its detection results. Chakchai *et al.* [19] believed that with the rapid development of the Internet, the number of network attacks has increased. Therefore, a model of network intrusion detection data mining classification was proposed. Hachmi and Limam [7] proposed a two-stage technology improved IDS based on the data mining algorithm and verified the performance of low false positive rate of the system. The experimental effect was significant. In this paper, IDS is taken as the research object to establish a data mining-based IDS model, the experimental results are obtained, and the relevant experimental conclusions are drawn. This study provides a reference for the research on the detection of network protection security vulnerability intrusion.

2 Data Mining

Data mining mainly refers to the process of searching for previously unknown but valuable and meaningful information through algorithms in a large number of data [18], which is an important operation step in knowledge-discovery in databases (KDD) [21] and relies mainly on artificial intelligence technology, statistics, *etc.* [11]. The main objectives of data mining include clas-

sification, clustering, prediction, bias analysis, *etc.* [16]. The main methods of data mining include mathematical statistical analysis, machine learning, *etc.* Data mining technology can mine normal or intrusive behavior patterns from large-scale audit data [23], where audit data is mainly composed of pre-processed and time-stamped audit records [17], and each audit record has some characteristics.

Data mining is widely used in various fields because of its own advantages [9], in which data mining is closely related to the computer field [5]. With the massive growth of network data information, network problems such as data information leakage, *etc.* have emerged one after another, so network security has become an arduous task. Therefore, it needs to combine some new ways to protect network security. In this paper, the detection performance of network protection security vulnerabilities intrusion is specifically studied by the method of data mining to verify its feasibility and practicality.

3 Detection of Network Protection Security Vulnerability Intrusion

3.1 IDS

The main working principle of IDS [15] is to perform correlation analysis on data information related to security in the network under the condition that the existing network performance is not affected, so as to detect intrusion behaviors. The role of IDS is [22] to identify illegal intrusion behaviors to perform corresponding response operations [14] and to detect system construction, weakness audit and user behaviors [2]. The main features of IDS include accuracy, scalability and fault tolerance [8]. In this study, IDS is taken as the main research object to carry out relevant simulation experiments, in which the network protection security vulnerability intrusion is mainly detected. Figure 1 shows the main components of IDS.

3.2 Research Based on Data Mining

In this study, IDS is taken as the main research object. The K-means clustering algorithm in data mining is used to detect network protection security vulnerability intrusion, and a model of IDS based on data mining is established. Firstly, through the corresponding collection system, the required data is selected as the initial clustering center. Then the relevant calculations are performed for each cluster center to obtain the relevant output results. Finally, though the output results obtained by the clustering calculation, the connection records are reasonably and scientifically assigned to distinguish the normal or abnormal connection records, and the relevant data is classified by the normal behavior pattern class and the abnormal behavior pattern class, thereby detecting the network protection security vulnerability invasion.

The main content of the K-means clustering algorithm is to use the similarity between the data through the iterative idea as a standard of the measure, to classify the objects into different similarity categories, making the internal similarity of each class high. In this algorithm, the solution of the cluster radius is inseparable from the data set itself. By extracting the distance characteristics of the data set itself, the appropriate cluster radius should be determined before clustering.

k represents the number of clusters, and the inaccuracy of the value of k will affect the quality of the final clustering results in this algorithm. Therefore, determining a suitable value of k is a major focus of the algorithm. When choosing the appropriate value of k, it needs to pay special attention to the two parameters of intra-class distance and inter-class distance. Specifically speaking, when the clustering effect is better, the intra-class distance is smaller, but the inter-class distance is larger. Therefore, in order to better balance the relationship between the two parameters, the method of linear combination is mainly adopted to carry out calculation and solution in this study.

In this algorithm, the Euclidean distance calculation method is used in this paper. It is assumed that the size of the data set is m, I indicates the number of iteration, $Z_j(I)$ represents the clustering center of category j, $X_i^{(j)}$ represents any data object in the class j, Z_j represents the new clustering center of class j, $I = 1$ is taken, and K initial clustering centers are selected, $Z_j(I)$, $j = 1, 2, \cdots, k$. The Euclidean distance between each data object and cluster center is calculated, $L(X_i, Z_j(I)) = \min\{(X_i, Z_j, (I)), i = 1, 2, \cdots, m, j = 1, 2, \cdots, K\}$. If the obtained Euclidean distance meets $L(X_i, Z_j(I)) = \min\{(X_i, Z_j(I)), j = 1, 2, \cdots, k\}$, $X_i \in W_k$ will be obtained.

The sum of the squares of the distances from all samples in the cluster domain to the cluster center is expressed as $H(c)$, δ indicates the iteration termination threshold, which needs to be determined whether it meets:

$$|H_c(I) - H_c(I-1)| < \delta. \tag{1}$$

If Equation (1) mentioned above is met, the algorithm will end, otherwise $I = I + 1$, k new cluster centers will be continued to calculate:

$$Z_j = \frac{1}{m} \sum_{i=1}^{m_j} X_i^{(j)}$$

The squared error criterion is also used in the algorithm, the sum of the squared errors of all samples in the data set is expressed as E, o represents the point in space, and m_i represents the average value of cluster C_i. E, the optimal result, can be defined as:

$$\min E = \min \sum_{i=1}^{k} \sum_{o \in C_i} |o - m_i|.$$

The detection performance of IDS is deeply analyzed through relevant parameters, wherein P_a, P_b, P_c indicate

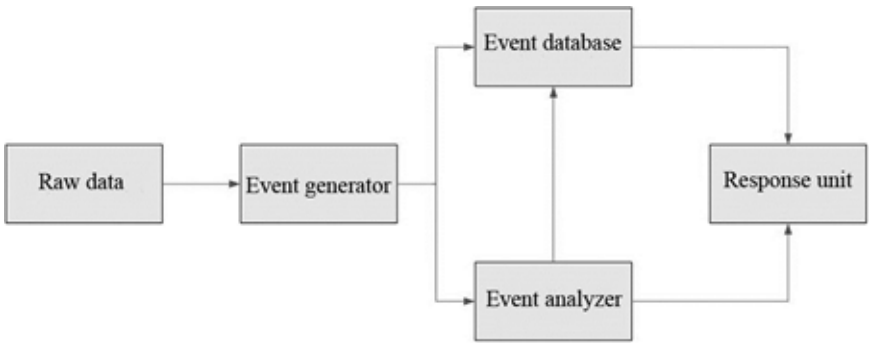


Figure 1: Main components of IDS

the detection rate, false negative rate, and false positive rate of the system respectively, N_d indicates the number of intrusion events detected correctly, M indicates the number of all intrusion events, M_{total} indicates the number of all events, N_e indicates the number of false negative intrusion events, and N_f indicates the number of false positive intrusion events. The formula shown below can be obtained:

$$P_a = \frac{N_d}{(}M) \times 100\%$$
$$P_b = \frac{N_e}{(}M) \times 100\%$$
$$P_c = \frac{N_f}{(}M_{total}) \times 100\%$$

4 Simulation Experiment

4.1 Experimental Methods and Parameters

In this study, the problem of the detection of network protection security vulnerability intrusion based on data mining is mainly researched. In this simulation experiment, the used related software is Matlab7.0, and the used experimental data set is KDD Cup 99, in which the test data includes 5000 pieces, and the training data includes 600 pieces. In the set of test data, normal data accounts for 75%, and vulnerability data accounts for 25%. Corresponding cluster analysis is performed on the data, and the vulnerability data in the experimental data is detected to obtain the output results of relevant parameters. At the same time, it is compared with the traditional IDS [13], the both methods are run six times respectively, the results of related parameters such as the detection rate of six different experiments are obtained, and the corresponding experimental conclusions are drawn for reference.

4.2 Experimental Results

(1) Cluster analysis Cluster analysis is performed on the data set of this simulation experiment, and the value of the cluster radius is adjusted to obtain the corresponding output results and experimental conclusions. Table 1 and Figure 2 can be obtained.

It can be seen from Table 1 that the total number of clusters shows a decreasing trend as the cluster radius increases. When the cluster radius is 1, the total number of clusters reaches the maximum value, i.e., 199. When the cluster radius is 10, the total number of clusters reaches the minimum value, i.e., 127. It can be obtained that the larger the cluster radius is, the smaller the total number of clusters is. Therefore, the cluster radius is inversely proportional to the total number of clusters, the more clusters are, the more detailed the cluster analysis is, and the smaller the false positive rate of IDS is.

It can be seen from Figure 2 that the cluster accuracy decreases as the cluster radius increases. When the cluster radius is 1, the cluster accuracy reaches the maximum value, i.e., 83.69%. When the cluster radius is 10, the cluster accuracy reaches the minimum value, 66.71%. It can be found that the smaller the cluster radius is, the better the accuracy of the algorithm is. Adjusting the cluster radius can effectively improve the cluster effect of the algorithm and have more obvious detection effect of vulnerability intrusion.

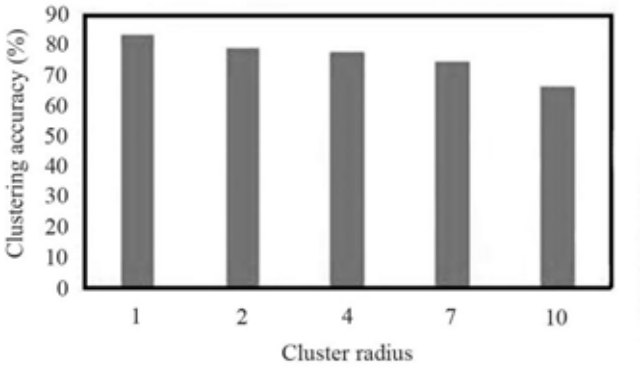


Figure 2: Comparison of the clustering accuracy

(2) Detection efficiency Figure 3 shows the detection

Table 1: The number of clusters

Cluster radius	Normal behavior pattern class	Abnormal behavior pattern class	Total number of clusters
1	1	198	199
2	0	196	196
4	2	189	191
7	6	172	178
10	11	116	127

time of the vulnerability data by IDS. It can be seen from Figure 3 that the detection time also shows a growing trend as the number of data increases. When the number of data reaches 700, the detection time of IDS reaches a maximum value, i.e., 16.5 s. The rate of increase shows a downward trend and gradually stabilizes although the time of detection is constantly increasing. Therefore, it can be obtained that the IDS using data mining has high detection efficiency for vulnerability data and remarkable experimental effect.

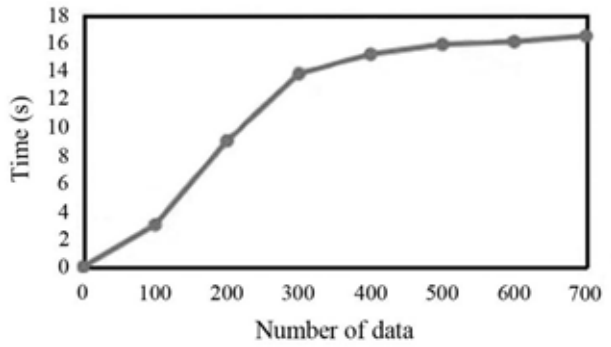


Figure 3: Detection time of vulnerability data by IDS

4.3 Comparative Analysis

In order to verify the performance of the IDS using data mining mentioned in this paper, it is compared with the traditional IDS, and Figure 4 and Table 2 are obtain. Figure 4 shows the comparative analysis of the relevant parameters between the two different methods in six experiments. In the six-time experiment, the IDS using data mining reaches the maximum value of the detection rate in the third experiment, i.e., 96.45% and the minimum value of the detection rate in the second experiment, i.e., 94.69%. The traditional IDS reaches the maximum value of the detection rate in the first experiment, i.e., 88.3% and the minimum value of the detection rate in the fifth experiment, i.e., 84.26%. Compared with the traditional IDS, the detection rate of the IDS using data mining is higher, and the experimental effect is more obvious. Therefore, it can be obtained that the ids using data mining has better detection performance, which is beneficial to the detection of network protection security and vulnerability intrusion.

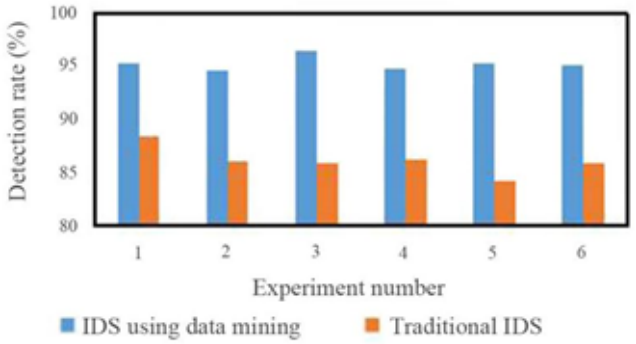


Figure 4: Detection rate in two different ways

It can be seen from Table 2 that the average detection rate of the traditional IDS is 87.75%, the average false negative rate is 28.49%, and the average false positive rate is 4.72%; while the average detection rate of the IDS using data mining is 95.63%, the average false negative rate is 20.23%, and the average false positive rate is 2.81%. Compared with the traditional IDS, the average detection rate of the IDS using data mining is higher, and the average false negative rate and average false positive rate are lower. Therefore, it can be obtained that the performance of the IDS using data mining is better, and the optimization effect on the experiment is more obvious. Therefore, in the future, the IDS using data mining has better development space and potential for the detection of network protection security vulnerability intrusion, but the traditional IDS needs continuous improvement and optimization.

5 Conclusion

Nowadays, network security issues are getting more and more attention. Therefore, in this paper, IDS is taken as the research object, and the corresponding simulation experiments are carried out. The obtained experimental results are as follows: the relationship between the cluster radius and the total number of clusters is inversely proportional; the smaller the cluster radius is, the better

Table 2: Comparison of the related parameters in two different ways

	Average detection rate (%)	Average false negative rate (%)	Average false positive rate (%)
IDS using data mining	95.63	20.23	2.81
Traditional IDS	87.75	28.49	4.72

the accuracy of the K-means clustering algorithm is; as the number of data increases, the detection time shows a growing trend, but the growth rate tends to be stable. At the same time, through comparing with the traditional IDS, the experimental results that the average detection rate of the IDS using data mining is higher than that of the traditional IDS, and the average false negative rate and false positive rate are lower are obtained. The experimental conclusion that the IDS using data mining has better detection performance for network protection security vulnerability intrusion is drawn. This study provides a new model for the research on the detection of network protection security vulnerability intrusion.

References

[1] A. A. Al-khatib, W. A. Hammood, "Mobile malware and defending systems: Comparison study," *International Journal of Electronics and Information Engineering*, vol. 6, no. 2, pp. 116–123, 2017.

[2] N. Y. Almusallam, Z. Tari, P. Bertok, *et al.*, "Dimensionality reduction for intrusion detection systems in multi-data streams — A review and proposal of unsupervised feature selection scheme," in *Emergent Computation*, pp. 467–487, 2017.

[3] M. H. R. Al-Shaikhly, H. M. El-Bakry, and A. A. Saleh, "Cloud security using Markov chain and genetic algorithm," *International Journal of Electronics and Information Engineering*, vol. 8, no. 2, pp. 96–106, 2018.

[4] F. J. Aparicio-Navarro, J. A. Chambers, K. Kyriakopoulos, *et al.*, "Using the pattern-of-life in networks to improve the effectiveness of intrusion detection systems," in *IEEE International Conference on Communications*, pp. 1–17, 2017.

[5] W. Chen, H. R. Pourghasemi, S. A. Naghibi, "Prioritization of landslide conditioning factors and its spatial modeling in Shangnan County, China using GIS-based data mining algorithms," *Bulletin of Engineering Geology and the Environment*, vol. 77, no. 2, pp. 611–629, 2017.

[6] B. Elhadj, W. Thomas, H. Walaa, "A critical review of practices and challenges in intrusion detection systems for IoT: Towards universal and resilient systems," *IEEE Communications Surveys & Tutorials*, vol. 20, no. 4, pp. 3496–3509, 2018.

[7] H. Fatma, L. Mohamed, "A two-stage technique to improve intrusion detection systems based on data mining algorithms," in *IEEE the 5th International Conference on Modeling, Simulation and Applied Optimization (ICMSAO'13)*, pp. 1–6, 2013.

[8] H. Hammami, H. Brahmi and S. Ben Yahia, "Security insurance of cloud computing services through cross roads of human-immune and intrusion-detection systems," in *The 32nd International Conference on Information Networking (ICOIN'18)*, pp. 174-181, 2018.

[9] C. Helma, T. Cramer, S. Kramer, *et al.*, "Data mining and machine learning techniques for the identification of mutagenicity inducing substructures and structure activity relationships of noncongeneric compounds," *Journal of Chemical Information and Modeling*, vol. 44, no. 4, pp. 1402–1411, 2004.

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

[11] B. Kang, J. Lijffijt, R. Santos-Rodríguez, *et al.*, "Subjectively interesting component analysis," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1615–1624, 2016.

[12] A. Keramatpour, A. Nikanjam, H. Ghaffarian, "Deployment of wireless intrusion detection systems to provide the most possible coverage in wireless sensor networks without infrastructures," *Wireless Personal Communications*, vol. 96, no. 3, pp. 1–14, 2017.

[13] M. Keshk, N. Moustafa, E. Sitnikova, *et al.*, "Privacy preservation intrusion detection technique for SCADA systems," in *IEEE Military Communications and Information Systems Conference (MilCIS'17)*, pp. 1–6, 2017.

[14] T. Miquel, J. Condomines, R. Chemali and N. Larrieu, "Design of a robust controller/observer for TCP/AQM network: First application to intrusion detection systems for drone fleet," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'17)*, pp. 1707-1712, 2017.

[15] N. Moustafa, J. Slay, "UNSW-NB15: A comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set)," in *Military Communications and Information Systems Conference (MilCIS'15)*, pp. 1–6, 2015. DOI: 10.1109/MilCIS.2015.7348942.

[16] R. Mousheimish, Y. Taher, K. Zeitouni and M. Dubus, "PACT-ART: Enrichment, data mining, and

complex event processing in the internet of cultural things," in *The 12th International Conference on Signal-Image Technology & Internet-Based Systems*, pp. 476–483, 2016.

[17] L. G. Noemí, A. S. Santiago, M. R. A. Blanca, M. O. M. Montserrat, T. Y. Chiang, "Divide and conquer! Data-mining tools and sequential multivariate analysis to search for diagnostic morphological characters within a plant polyploid complex (Veronica subsect. Pentasepalae, Plantaginaceae)," *Plos One*, vol. 13, no. 6, pp. e0199818, 2018.

[18] G. Omid, R. Hashem, B. Thomas, *et al.*, "A new GIS-based data mining technique using an adaptive neuro-fuzzy inference system (ANFIS) and k-fold cross-validation approach for land subsidence susceptibility mapping," *Natural Hazards*, vol. 94, no. 2, pp. 497–517, 2018.

[19] C. So-In, N. Mongkonchai, P. Aimtongkham, K. Wijitsopon and K. Rujirakul, "An evaluation of data mining classification models for network intrusion detection," in *International Conference on Digital Information & Communication Technology & Its Applications*, pp. 90-94, 2014.

[20] B. Subba, S. Biswas, S. Karmakar, "Enhancing performance of anomaly based intrusion detection systems through dimensionality reduction using principal component analysis," in *IEEE International Conference on Advanced Networks & Telecommunications Systems*, pp. 1-6, 2016.

[21] K. R. Thorp, G. Wang, K. F. Bronson, *et al.*, "Hyperspectral data mining to identify relevant canopy spectral features for estimating durum wheat growth, nitrogen status, and grain yield," *Computers and Electronics in Agriculture*, vol. 136, pp. 1–12, 2017.

[22] S. Vakili, J. M. P. Langlois, B. Boughzala, *et al.*, "Memory-efficient string matching for intrusion detection systems using a high-precision pattern grouping algorithm," in *Proceedings of Symposium on Architectures for Networking and Communications Systems (ANCS'16)*, pp. 37–42, 2016.

[23] Q. Wang, G. Z. Yao, G. M. Pan, *et al.*, "Analysis of on medication rules for Qi-deficiency and blood-stasis syndrome of chronic heart failure based on data mining technology," *China Journal of Chinese Materia Medica*, vol. 42, no. 1, pp. 182–186, 2017.

Biography

Jinming Zhang born in Yantai, male, from Yantai, Shandong, China, has gained the master's degree. He is now working in Yantai vocational college. He is an associate professor and senior engineer. He is interested in computer network technology, information security technology and cloud computing technology.

Quadrivium: A Trivium-Inspired Pseudorandom Number Generator

Latoya Jackson[1] and Yeşem Kurt Peker[2]
(Corresponding author: Yeşem Peker)

TSYS School of Computer Science, Columbus State University[1]
Columbus, GA, USA
TSYS School of Computer Science, Columbus State University[2]
Columbus, GA, USA
(Email: peker_yesem@columbusstate.edu)
(Received July 4, 2018; Revised and Accepted Oct. 28, 2018; First Online Feb. 2, 2019)

Abstract

A pseudorandom number generator (PRNG) is an algorithm that produces seemingly random number sequences. They are employed in applications requiring randomness such as arbitrary sample selection in statistical sampling and secret key generation for ciphers. Where unpredictability is a concern, a cryptographically secure PRNG (CSPRNG) is the only type of PRNGs suitable for such applications. CSPRNGs are specially designed to withstand security attacks. In this paper, after describing a well-known lightweight stream cipher Trivium, we present Quadrivium, a PRNG inspired by the design of Trivium. We compare the statistical properties of Quadrivium by that of Trivium using NIST Statistical Test Suite and Dieharder: A Random Number Test Suite. The analyses show that Quadrivium performs as well as Trivium and has the advantage of producing longer sequences of random bits.

Keywords: Crytographically Secure Pseudorandom Number Generator; CSPRNG; PRNG; Pseudorandom Number Generator

1 Introduction

A random number generator is an object that produces number sequences emulating characteristics of truly random sequences. They are relevant in statistical sampling, Monte Carlo simulation, gaming, internet gambling, cryptography as well as other areas in need of random values. In statistical sampling, generators are used to select arbitrary samples for analysis. Monte Carlo simulation methods employ RNGs to solve optimization, numerical integration and probability distribution problems. Computer-controlled characters and procedural generation in electronic gaming use generators as a source of randomness. Internet gambling as well requires this same type of source to ensure game integrity and combat cheating. Randomness is also implemented to generate secret keys for well-known ciphers such as AES, RSA and Blowfish; It is used to encrypt messages for One Time Pads or to conceal information in protocols by converting the data to seemingly random sequences.

There are two approaches used to generate random sequences. One is a truly random number generator (TRNG), which outputs strings of random quantities using an unpredictable physical source. The other approach is a pseudorandom number generator (PRNG) which uses deterministic methods to generate "random" sequences. Various methods for generating pseudorandom numbers are being proposed and studied such as [3, 8, 9].

PRNGs are considered more suitable for computing devices in comparison to its genuine counterpart. They are portable; Do not consume a lot of resources (in terms of memory); And operate on a wide range of devices. However, the deterministic nature of the generation process is a concern. PRNGs should be carefully tested to verify that their output approximates a sequence of true random numbers. There is no single test available that can determine if a PRNG generates numbers that have the characteristics of randomness. The best that can be done is to assess a PRNG via a series of tests. A PRNG must perform well on multiple tests to be considered random.

The basic construct of a PRNG is a seed, a generating algorithm and an output. The seed is from a finite set of seeds and is typically a truly random number. It is used to initialize the generator. The generating algorithm has an internal state comprised of all stored values, parameters and variables the generator relies on to function. Additionally, it possesses an update function to refresh the internal state as well as an output function which yields a pseudorandom output. Like the seed, the output is an element of a finite set of possible outputs and an elongated transformation of the seed.

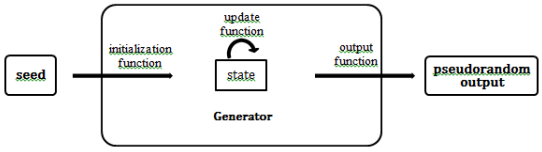


Figure 1: Diagram of a pesudorandom number generator

2 CSPRNGs

2.1 Definition and Properties

In addition to random-like statistical properties, a cryptographically secure PRNG (CSPRNG)—unlike a non-cryptographic PRNG—must possess the property of unpredictability. Unpredictability guarantees pseudorandom values produced by a generator lacks structure, cannot be controlled nor conform to some pattern. Unpredictability does not equate to true randomness. It is another form of randomness that requires high entropy. It is considered more practical than perfect randomness—which is not accessible for all systems. The degree of unpredictability in the "random" generation process directly affects the strength of the cryptographic algorithm. In cases of insufficient unpredictability, generators are susceptible to attacks. By definition, a CSPRNG is unpredictable if the next output value in a sequence is computationally infeasible even if a sequence of previous output values is known [10]. This is formally termed as the next-bit test.

2.2 PRNG Failures

Some PRNG failures are attributed to the lack of entropy or acquisition of entropy within a generating environment. Entropy is a collection of sources employed to seed, and for some PRNGs, update its internal state. Examples of entropy sources include mouse movement, keystroke timing and noise from a computing system's soundcard. Other failures may be based on short periodicity or linearity of the generating function. Low entropy, short periodicity, and linearity in a PRNG facilitate the prediction of the generated numbers in a feasible amount of time. As a consequence, the generator becomes vulnerable to attacks.

Debian experienced a security breach with its OpenSSL distribution. The pseudorandom generator included in the implementation was incapable of acquiring high levels of entropy. This caused the PRNG to produce 32,767 private keys. The small key space ensue highly predictable keys. Other Debian-based products, like Ubuntu, were affected by this PRNG failure.

Another case involves the internationally used MIFARE Classic chip. It has applications in contactless smart cards and proximity cards. In a 2008 paper by de Koning Gans, Hoepman and Garcia, the researchers were able to recover keystreams, read memory blocks and modify memory blocks from the chip. This was all due to the low entropy collecting PRNG implemented in MIFARE [5].

Security Socket Layer (SSL) uses a PRNG to generate a random key. The key is used in a cryptographic algorithm to encrypt information flowing between client and server. Netscape utilized its own implementation of SSL to protect transmission of sensitive data over its browser. However, two computer science students were able to decipher encrypted messages sent over Netscape Web by exploiting the flaws in the PRNG used in the Netscape SSL implementation. The flaw was due to poor seeding of the generator. Even though unique, the seed values taken from the running system (process ID, parent process ID and time of day) were predictable. Hence, the key was retrievable as well as the messages [6].

Shortly following a publication which analyzed the security of popular SecureRandom constructs, a Bitcoin incident occurred leaving its Android users vulnerable to theft [7]. The two events are related in that SecureRandom is a special PRNG for cryptographic applications and Android uses it for cryptographic Bitcoin procedures. However, the Android SecureRandom implementation had a bug that caused the generator to yield predictable sequences. The paper revealed how the generator produced colliding values—making the private key recoverable. The paper also discussed the PRNG's defects in entropy collection and the capability to overwrite the seed value.

3 Lightweight Cryptography and Trivium

Lightweight cryptography is a cryptographic protocol or algorithm intended for usage within constrained device networks. Constrained devices are objects that have limited processing power, memory storage capabilities, and power resources. Typically, they do not possess the proper resources needed to employ traditional cryptographic algorithms. There are some cases where traditional algorithms can be implemented but it is accompanied with significant performance reduction. (Performance encompasses power and energy consumption as well as latency and throughput.) Lightweight cryptography provides a solution for the performance-security tradeoff problem that exists for compact devices.

Trivium, designed by Christophe DeCanniere and Bart Preneel, is a stream cipher intended to operate in constrained spaces. It was selected for the eSTREAM portfolio of lightweight stream ciphers for hardware application. Trivium is also efficient in software-based environments. Additionally, it has been designated by International Organization for Standardization (ISO) as a keystream generator for lightweight stream ciphers [1]. Its keystream may be used as a source for pseudorandom bits.

3.1 Structure of Trivium

Trivium can be described as a bit-oriented stream cipher conducting operations at the bit level. The internal state of the cipher consists of three registers totaling to 288 bits. The first register holds 93 state bits, the second holds 84 state bits and the last register holds 111 state bits. The algorithmic component is broken down into two phases, the setup phase and the generation phase (which is also responsible for updating the internal state of the cipher). Trivium takes in a key and IV of 80 bits each and guarantees to generate up to 2^{64} keystream bits [4].

When creating Trivium, the authors had two mandatory specifications the construction must contain. First, the structure must generate seemingly uncorrelated keystreams. Second, the construction must also be efficient such that there is a high throughput of generated keystream bits per cycle per logic gate. The authors referenced the operations of block ciphers as a solution to their specifications. In comparison to stream ciphers, block ciphers are more developed. Many techniques have been uncovered to bolster the efficiency of block ciphers to operate speedily and with low space consumption. Additionally, the security of a block cipher is well researched and understood.

3.2 Trivium's Algorithm

Trivium requires an 80-bit key and 80-bit initialization vector for set up. Initialization begins with the key being copied to the first register. After copying the key to the first 80 slots, the remaining state bits (denoted as s) are set to zero. The initialization vector is then written to the second shift register. The rightmost four bits in this register are set to zero. The last register has all its bits set to zero except for the last three bits; They are set to one. The internal state is refreshed 1152 times to ensure that all bits are influenced by the key and the IV. The pseudocode is given below in (Algorithm 1).

Algorithm 1 Initialization of Trivium

Begin
Initialize *registers.*
for i = 1 to 1152 **do**
  $t_1 \leftarrow s_{66} + s_{91} \cdot s_{92} + s_{93} + s_{171}$
  $t_2 \leftarrow s_{162} + s_{175} \cdot s_{176} + s_{177} + s_{264}$
  $t_3 \leftarrow s_{243} + s_{286} \cdot s_{287} + s_{288} + s_{69}$
  $[s_1, s_2, \cdots, s_{93}] \leftarrow [t_3, s_1, \cdots, s_{92}]$
  $[s_{94}, s_{95}, \cdots, s_{177}] \leftarrow [t_1, s_{94}, \cdots, s_{176}]$
  $[s_{178}, s_{279}, \cdots, s_{288}] \leftarrow [t_2, s_{178}, \cdots, s_{287}]$
end for

The Trivium generation process actually begins by performing an exclusive or operation on two specific bits from each register. The resulting three bits collectively undergo another exclusive or operation. The result from this last step is a single bit that is added to the keystream. The pseudocode is given below in (Algorithm 2).

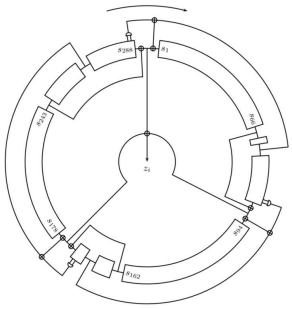


Figure 2: Structure of Trivium

Algorithm 2 Generation, Update & Output Algorithm for Trivium

m = requested bits
for i = 1 to m **do**
  $t_1 \leftarrow s_{66} + s_{92}$
  $t_2 \leftarrow s_{162} + s_{177}$
  $t_3 \leftarrow s_{243} + s_{288}$
  $z_i \leftarrow t_1 + t_2 + t_3$
  {Trivium Updates by doing the following:}
  $t_1 \leftarrow t_1 + s_{91} \cdot s_{92} + s_{171}$
  $t_2 \leftarrow t_2 + s_{175} \cdot s_{176} + s_{264}$
  $t_3 \leftarrow t_3 + s_{286} \cdot s_{287} + s_{69}$
  $[s_1, s_2, \cdots, s_{93}] \leftarrow [t_3, s_1, \cdots, s_{92}]$
  $[s_{94}, s_{95}, \cdots, s_{177}] \leftarrow [t_1, s_{94}, \cdots, s_{176}]$
  $[s_{178}, s_{279}, \cdots, s_{288}] \leftarrow [t_2, s_{178}, \cdots, s_{287}]$
end for

Trivium produces only a single bit at a time. This entire process is reiterated until the desired length is reached. The pseudorandom output can be simplified to:

$$z_i \leftarrow s_{66} + s_{93} + s_{162} + s_{177} + s_{243} + s_{288}.$$

The unpredictability of the z_i is dependent on the constant rotation of the state bits and transformation of some bits with each state update. An attacker must be aware of the internal state to accurately predict the next bit. This is quite difficult given that each Trivium state is constructed to be linearly independent.

4 Quadrivium

Quadrivium is a pseudorandom number generator designed with a software implementation in mind. The structure is primarily modeled after Trivium but coalesce findings in [11]; The definition of primitive polynomials; And feedback functions found in linear feedback shift registers (LFSRs).

4.1 Design

Quadrivium is a 384 bit state PRNG. The generator is partitioned into four registers of 98-bit, 97-bit, 95-bit and

94-bit length. It requires a total of 160 random bits for initialization. To understand the design principles of Quadrivium, first, we will review some standard definitions relating to polynomials.

A polynomial p(x) is a mathematical expression consisting of a sum of terms where each term includes x raised to a non-negative integer power and multiplied by a coefficient. It can be written as:

$$p(x) := \sum_{i=0}^{n} a_i x^i. \tag{1}$$

The variable a_i denotes the coefficient of x_i and an is different from zero. The value of n is the degree of $p(x)$. For this discussion, we restrict all $p(x)$ to polynomials in GF(2). Therefore, all coefficients are either zero or one. A polynomial $p(x)$ is said to be trivial if the degree of $p(x)$ is $-\infty$, indicating a zero polynomial, or 0, a constant polynomial; Otherwise, it is nontrivial. A polynomial $p(x)$ is an irreducible polynomial if it cannot be factored into two or more non-trivial polynomials.

A primitive polynomial is an irreducible polynomial with degree $n > 0$ and a polynomial order (or period) of $2^n - 1$. A linear feedback shift register (LFSR) is an object that employs the function $f : 0, 1^n \rightarrow 0, 1$ such that the output bit x_0 is:

$$x_0 = \sum_{i=0}^{n} a_i x^i. \tag{2}$$

The variable a_i is a coefficient $\epsilon 0, 1$. The feedback function has a period of $2^n - 1$. It is a common practice to employ primitive polynomials as a feedback function.
In [11], the authors noted the following as the active bits in Trivium:

$$\{s_{66}, s_{69}, s_{93}\}\ \{s_{162}, s_{171}, s_{177}\}\ \{s_{243}, s_{264}, s_{288}\}.$$

Recognizing that each index is a multiple of 3, these bits can be generalized as

$$\{s_{a_{m_1}}, s_{a_{m_2}}, s_{a_{n_1}}\}\ \{s_{a_{m_3}}, s_{a_{m_4}}, s_{a_{n_2}}\}\ \{s_{a_{m_5}}, s_{a_{m_6}}, s_{a_{n_3}}\}.$$

For the next part of the discussion, we are only concerned with the family of variables $\{m_1, m_2, n_1\}$, $\{m_3, m_4, n_2\}$ and $\{m_5, m_6, n_3\}$. If we consider these variables as powers of x for non-zero terms in a polynomial, we get the following:

$$x^{m_1} + x^{m_2} + x^{n_1} + x^{m_3} + x^{m_4} + x^{n_2} + x^{m_5} + x^{m_6} + x^{n_3}.$$

Let $k\epsilon\mathbb{N}$ and $q(x)$ is a primitive polynomial. A polynomial $p(x)$ is a k-order primitive polynomial if

$$p(x) = (x + 1)^k q(x). \tag{3}$$

According to [11], Trivium is a 3-order primitive polynomial with

$$q(x) = x^{22} + x^{23} + x^{31} + x^{54} + x^{57} + x^{59} + x^{81} + x^{88} + x^{96}. \tag{4}$$

We were motivated by this definition to extend Trivium to a k^{th} round and use primitive polynomials to select the active state bits in Quadrivium. Our construction differentiates from the one proposed in [11] in two major respects. First, Quadrivium is driven by the principles of PRNGs. This results in a pseudorandom sequence of lesser correlated bits. Second, the active state bits were redefined to be in agreement with the concept of PRNGs. Since Quadrivium takes a PRNG approach, our concern lies in the linearity of the pseudorandom output. We imposed several criteria to be in accordance with this approach. Recall in Trivium that the pseudorandom bit z_i is the sum of state bits 66, 92, 162, 177, 242 and 288.

In our construction, we redefine the active state bits to those responsible for pseudorandom bit z_i. Second, the active state bits must be derived from a primitive polynomial of degree 384. In Trivium, two state bits from each register is used to generate a single bit output. This should be the criterion to restrict the number of active state bits. As a result, the active state bits are s_{49}, s_{98}, s_{147}, s_{195}, s_{243}, s_{290}, s_{337} and s_{384}.

4.2 Algorithm

The initialization procedure, like Trivium, uses an 80-bit key and 80-bit IV. For the first and second registers, the key and IV is copied to the registers, respectively. The third register is filled with zeroes excluding the last three state bits. Those are set to one. The last register is initialized with one-bit bit values except for the last four bits. They are zeroes. Once the registers are loaded, the rotate procedure is executed. The rotate procedure is reiterated for four full cycles (Algorithm 3).

Algorithm 3 Initialization of Quadrivium

1: Begin
2: Initialize *registers.*
3: **for** i = 1 to 1536 **do**
4: $t_1 \leftarrow s_{49} + s_{96} \cdot s_{97} + s_{98} + s_{171}$
5: $t_2 \leftarrow s_{147} + s_{193} \cdot s_{194} + s_{195} + s_{358}$
6: $t_3 \leftarrow s_{243} + s_{288} \cdot s_{289} + s_{290} + s_{69}$
7: $t_4 \leftarrow s_{337} + s_{382} \cdot s_{383} + s_{384} + s_{264}$
8: $[s_1, s_2, \cdots, s_{98}] \leftarrow [t_2, s_1, \cdots, s_{97}]$
9: $[s_{99}, s_{100}, \cdots, s_{195}] \leftarrow [t_4, s_{99}, \cdots, s_{194}]$
10: $[s_{196}, s_{197}, \cdots, s_{290}] \leftarrow [t_1, s_{196}, \cdots, s_{289}]$
11: $[s_{291}, s_{292}, \cdots, s_{384}] \leftarrow [t_3, s_{292}, \cdots, s_{383}]$
12: **end for**

The main procedure for Quadrivium is quite similar to Trivium (Algorithm 4). Unlike Trivium, we did not include the previous values of t_i in the update function to produce the current values of t_i. The inclusion of the values does not necessarily have a negative impact on the generator. The decision to exclude these values was to restrict their influence to only the output bit versus both the output and the new state bits in Trivium. Both Quadrivium and Trivium has a nonlinear internal state

Algorithm 4 Generation, Update & Output Algorithm for Trivium

m = requested bits

for i = 1 to m **do**
  $t_1 \leftarrow s_{49} + s_{98}$
  $t_2 \leftarrow s_{147} + s_{195}$
  $t_3 \leftarrow s_{243} + s_{290}$
  $t_4 \leftarrow s_{337} + s_{384}$
  $z_i \leftarrow t_1 + t_2 + t_3 + t_4$
  $t_1 \leftarrow s_{96} \cdot s_{97} + s_{171}$
  $t_2 \leftarrow s_{193} \cdot s_{194} + s_{358}$
  $t_3 \leftarrow s_{288} \cdot s_{289} + s_{69}$
  $t_4 \leftarrow s_{382} \cdot s_{383} + s_{264}$
  $[s_1, s_2, \cdots, s_{98}] \leftarrow [t_2, s_1, \cdots, s_{97}]$
  $[s_{99}, s_{100}, \cdots, s_{195}] \leftarrow [t_4, s_{99}, \cdots, s_{194}]$
  $[s_{196}, s_{197}, \cdots, s_{290}] \leftarrow [t_1, s_{196}, \cdots, s_{289}]$
  $[s_{291}, s_{292}, \cdots, s_{384}] \leftarrow [t_3, s_{292}, \cdots, s_{383}]$
end for

so it is difficult to determine their periodicity. In [4], the authors noted that the period of Trivium is at least $2^{96-3} - 1$. This is under the assumption that the state evolves linearly. For Quadrivium, the period is at least $2^{384} - 1$, given the same assumption. This is based on the fact that the output function is derived from a primitive polynomial.

5 Statistical Testing

Statistical testing is one of the most common methods used to determine the output quality of PRNGs. In this section we provide brief descriptions of the three well-known statistical testing suits; Namely NIST Statistical Test Suite, Diehard Battery of Tests, and the Dieharder Random Number Test Suite.

5.1 NIST Statistical Testing Suite

NIST Statistical Test Suite (STS) for cryptographically secure RNGs and PRNGs is a standard for statistical testing. The suite contains fifteen tests which analyze the quality of a PRNG's output; And determine whether the outputs mimic the behaviors of truly random sequences. Each test uses a test statistic to determine whether to reject the null hypothesis or not. The null hypothesis is the tested sequence is random; It lacks a pattern and portrays irregularity. The alternative hypothesis is the sequence is not random, a pattern was detected therefore it is predictable [2].

The assessments focus on different behaviors which indicate predictability in a sequence; They can be classified into four main types. The first type is frequency tests. They are the Frequency test (Freq), Frequency Test within a Block (Block), Runs Test (Runs), Test for the Longest Run of Ones in a Block (Long). The following two tests, Binary Matrix Rank Test (Rank) and Discrete Fourier Transform Test (FFT), fall under the repetitive patterns type. Non-overlapping Template Matching Test (NOTemp), Overlapping Template Matching Test (OTemp), Maurer's "Universal Statistical" Test (Univ), Linear Complexity Test (LinCom), Serial Test (Serial) and Approximate Entropy Test (AppEnt) are pattern matching types. The fourth type is random walks and consists of Cumulative Sums Test (CuSum), Random Excursions Test (RanEx) and Random Excursions Variant Test (RanExV) [12].

Even though all tests focus on different aspects of an ensemble, there are three assumptions that they all hold about random outputs. These assumptions are taken in consideration when determining the quality of a PRNG's outputs and if they are comparable to a set of truly random sequences. The assumptions are uniformity, scalability and consistency. Looking at a random sequence of length n, uniformity means the occurrence of zeroes should be one-half of the sequence, likewise the occurrence of ones. Scalability determines to what degree is a sequence random. This property also expects that all subsets of a random sequence must also be random. Consistency expresses the behavior of a PRNG. According to the literature, a consistent PRNG will always produce the random sequences of equal quality. It is not necessary to conduct all tests in the suite when analyzing a PRNG. The analyst is responsible for selecting the appropriate combination of assessments used to study a generator [2].

5.2 Diehard Battery of Test

Diehard is a statistical testing suite created by George Marsaglia, who is also the creator of pseudorandom number generator Xorshift. Diehard includes sixteen tests—fifteen personally authored by Marsaglia—that gauge the randomness quality of a generator. The tests require a binary file of at least 80 million random bits as input. The number of bits needed for to execute each test varies.

A majority of the assessments in the suite uses a p-value to determine if a sequence is random. This is similar to the NIST STS which also has a number of tests that rely on p-values to draw a conclusion. In statistics, p-values represent the probabilities that some arbitrary event will occur; Their purpose is to accept or reject the null hypothesis, which is the tested claim. In Diehard, the null hypothesis is the analyzed sequence is random. Tested sequences are acknowledged as random if p-values are not close to zero or one. Contrarily, in NIST STS, the further the p-value is to one, the further a sequence is to being truly random.

Another difference between the two testing suites is the analysis of the results. NIST STS specifies the p-value needed to reject the null hypothesis. Diehard battery of tests is ambiguous and only states that the p-values should be uniform on the set $[0, 1)$.
The names of the exams included in the Diehard battery of tests are Birthday Spacings Test; Overlapping 5-

Permutation Test; Separate Binary Rank Tests for 31x31, 32x32 and 6x8 matrices; Bitstream Test; OPSO, OQSO and DNA (Overlapping Pairs Sparse Occupancy, Overlapping Quadruples Sparse Occupancy and DNA Test, respectively); Separate Count the 1s Test for byte-streams and specific bytes; This is a Parking Lot Test; Minimum Distance Test; 3DSpheres Test; Squeeze Test; Overlapping Sums Test; Runs test—which is a standard test; And Craps Test.

5.3 Dieharder: A Random Number Test Suite

Dieharder is a test pack for random number generators. The suite includes modified tests from Diehard battery of tests, NIST Statistical Test Suite as well as some assessments created by Robert G. Brown, the chief developer of Dieharder. The test suite is an open source project whose purpose is to become a one-stop source for quantifying randomness. The project encourages inclusion of other new testing schemes from other developers. Dieharder is primarily concerned with analyzing the randomness quality and speed of truly random and pseudorandom number generators.

In comparison to STS and Diehard, the suite prefers to examine the actual generator, not a random output file produced by the generator. The reasoning behind this is "perfect randomness is the production of 'unlikely' sequences of random number at an average rate." Looking at the output alone is not sufficient to declare randomness; The likelihood of the sequence as a whole cannot be determined. Even though Dieharder prefers the aforementioned method of testing, it can still accommodate file-based inputs.

In Dieharder, parameters from STS and Diehard are altered so failures are concluded without ambiguity. Moreover, the Diehard tests are improved in three ways. One, assessments that uses KSTEST, Kolmogorov-Smirnov test, imposes a higher default quantity of one hundred p-values. This coincides with Dieharder's aim to determine unambiguous failure. Two, analysts have more control over tests that use samples. Sample sizes are treated as a variable rather than a fixed constant. Three, assessments that employs overlapping techniques on sequences were adjusted to use non-overlapping techniques. Please note that these improvements were made only if it was possible.

There are ten additional tests in Dieharder .They were created by Robert G. Brown and are called RGB. They are the following: Bit Distribution Test, Generalized Minimum Distance Test, Permutations Test, Lagged Sums Test, KSTest (Kolmogorov-Smirnov Test) Test, DAB Byte Distribution Test, DCT (Frequency Analysis), DAB Fill Tree Test, DAB Fill Tree 2 Test and DAB Monobit Test.

6 Results

We employed NIST STS and Dieharder: A Random Number Test Suite to assess the performance of Trivium and Quadrivium. We used Trivium as a benchmark for the performance of the Quadrivium. For all analyses, three different pseudorandom data files from each generator were tested. Each file consisted of 122.88 million bytes. This was determined by the Dieharder test suite which requires about 31 million integers for proper analysis.

6.1 STS Results

All tests in the suite were conducted on each file. For testing purposes the data file was segmented into 700 subsequences, each one million bits in length. The significance level, $\alpha = 0.01$, determined the number of subsequences used. For this level, at least one hundred sequences must be available for testing. The subsequence length was chosen based on the Maurer's "Universal Statistical" Test. This assessment requires approximately 1.4 million bit-long sequences to evaluate a generator correctly. This is the largest quantity amongst all the tests in the suite.

For each STS run, the suite returns twelve values for each test. One value is the proportion of subsequences passing the respective test. Another value is a single p-value of all the p-values determined. The remaining values are the distribution of p-values over ten subintervals on $(0, 1]$. The p-value is used to determine the degree of uniformity amongst sequences. The closer a p-value is to one; The closer it is to perfect uniformity. A p-value $\geq$ 0.0001 and a proportions value of 0.978 are required to pass a test.

In Tables 1 and 2, 'P-val' denotes the p-value and 'Prop' stands for proportion. The Cumulative Sums test and the Serial Test assess in two directions, forward and backward. In the aforementioned tables, 'F' and 'B' signifies the results for the forward and backward direction, respectively. The Non-Overlapping Template, Random Excursion and Random Excursion Variant Test provide multiple sets of test results. The proportion values shown reflect the average of these tests' results.

Quadrivium outperformed Trivium on the Frequency within a Block, Tests for the Longest Runs, Overlapping Template Matching, Maurer's "Universal Statistical" Test and Linear Complexity Tests. Quadrivium had the highest average proportions for the Frequency within a Block, Tests for the Longest Runs, Overlapping Template Matching, Maurer's "Universal Statistical" Test and Linear Complexity Tests.

6.2 Dieharder Results

Diehard Battery of Tests and RGB tests were conducted under the Dieharder test suite. Each dataset was parsed into unsigned 32 bit integers totaling 30.72 million integers. The suite returned two results: a p-value and an assessment of passed, weak or failed. A weak assessment

signifies the p-value ≤ 0.005. A failed assessment signifies the p-value ≤ 0.000001.

Tables 3 and 4 show the assessment counts for all collected data. Even though Diehard and RGB are sets of fifteen and ten tests, respectively, the total assessment counts are greater. This is attributed to the fact that some of the tests are conducted with multiple parameters. One of the RGB tests, Lagged Sums Test, for example, has 33 different variants.

Both generators had one data set that was considered weak for the OPSO test, Trivium dataset 1 and Quadrivium dataset 2. The Binary Matrix Rank Test 32x32 was also a common problem for the generators. Trivium failed this test with dataset 1 and was considered weak for dataset 3 while Quadrivium received a weak assessment for data sets 1 and 2. The other weak assessments are as follows: Trivium dataset 1, Craps 2 test; Trivium dataset 3, Count the ones test for bytes; Quadrivium dataset 2, Runs test; Quadrivium dataset 3, OQSO.

7 Conclusion

In this paper, we presented a revised model of Trivium that focused on improving the selection of state bits. We considered the entire state of Trivium to make improvements versus its individual registers in previous works. We were aware that the unpredictability of pseudorandom sequences is directly correlated to the entire set of state bits selected to yield stream bits and proposed a solution in our model.

The analyses we presented indicates that Quadrivium exhibits more characteristics of uniformity than Trivium. From Tables 1 and 2, we see that Quadrivium has more p-values closer to one than Trivium. Tables 3 and 4 shows us that Quadrivium has a higher passing rate on the Diehard and RGB tests. Given the data from all the tables, we can conclude that Quadrivium consistently performs well on tests that checks for linear complexity, pattern matching and pseudorandomness on a sequence-level.
Future work can be to improve Quadrivium such that it is seemingly random on a bit level. Potential research could also be to determine the period and security of Quadrivium.

References

[1] "Information technology — security techniques — lightweight cryptography — part 3: Stream ciphers,". Tech. Rep. ISO/IEC 29192-3, October 2012.

[2] et al A. Rukhin. "A statistical test suite for random and pseudorandom number generators for cryptographic applications,". Tech. Rep. NIST Special Publication 800-22 Revision 1a, April 2010.

[3] K. Szczypiorski B. Stoyanov and K. Kordov, "Yet another pseudorandom number generator," *Interna-*

Table 1: STS results of Quadrivium datasets

Tests	**Dataset 1**		**Dataset 2**		**Dataset 3**	
	P-val	*Prop*	*P-val*	*Prop*	*P-val*	*Prop*
Freq	0.074	0.986	0.577	0.988	0.009	0.991
Block	0.937	0.989	0.640	0.987	0.702	0.988
CuSum F	0.243	0.988	0.971	0.985	0.486	0.991
CuSum B	0.435	0.983	0.613	0.987	0.458	0.991
Runs	0.011	0.986	0.219	0.985	0.138	0.994
Long	0.211	0.992	0.341	0.994	0.218	0.990
Rank	0.911	0.990	0.426	0.978	0.634	0.995
FFT	0.174	0.981	0.955	0.988	0.013	0.982
NOTemp	—	—	—	0.989	—	0.990
OTemp	0.460	0.986	0.762	0.988	0.534	0.990
Univ	0.511	0.981	0.795	0.985	0.944	0.981
AppEnt	0.893	0.992	0.713	0.990	0.672	0.991
RanEx	—	—	—	0.990	—	0.991
RanExV	—	—	—	0.992	—	0.990
Serial F	0.559	0.987	0.957	0.992	0.622	0.991
Serial B	0.984	0.980	0.308	0.992	0.768	0.990
LinCom	0.229	0.991	0.719	0.992	0.592	0.988

Table 2: STS results of Trivium datasets

Tests	**Dataset 1**		**Dataset 2**		**Dataset 3**	
	P-val	*Prop*	*P-val*	*Prop*	*P-val*	*Prop*
Freq	0.277	0.998	0.291	0.992	0.374	0.994
Block	0.341	0.985	0.722	0.988	0.034	0.991
CuSum F	0.138	0.995	0.634	0.991	0.756	0.995
CuSum B	0.332	0.994	0.899	0.992	0.352	0.991
Runs	0.944	0.988	0.155	0.990	0.450	0.997
Long	0.876	0.994	0.534	0.987	0.348	0.984
Rank	0.210	0.992	0.264	0.991	0.223	0.984
FFT	0.592	0.991	0.158	0.990	0.474	0.980
NOTemp	—	0.989	—	0.989	—	0.983
OTemp	0.323	0.985	0.366	0.990	0.187	0.987
Univ	0.098	0.995	0.795	0.987	0.208	0.992
AppEnt	0.146	0.988	0.705	0.990	0.563	0.990
RanEx	—	0.991	—	0.988	—	0.990
RanExV	—	0.992	—	0.989	—	0.987
Serial F	0.023	0.987	0.352	0.987	0.932	0.985
Serial B	0.453	0.991	0.273	0.984	0.376	0.984
LinCom	0.657	0.994	0.260	0.991	0.000	0.978

Table 3: Diehard tests assessment counts

Assessment	**Quadrivium Datasets**			**Trivium Datasets**		
	1	2	3	1	2	3
Passed	18	16	18	16	19	17
Weak	1	3	1	2	0	2
Failed	0	0	0	1	0	0

Table 4: RGB tests assessment counts

Assessment	Quadrivium Datasets			Trivium Datasets		
	1	2	3	1	2	3
Passed	46	42	42	36	43	30
Weak	6	9	4	16	6	18
Failed	9	10	15	9	12	13

tional Journal of Electronics and Telecommunications, vol. 63, no. 2, pp. 195–199, 2017.

[4] C.De Canniere and B.Preneel, "Trivium," *Lecture Notes in Computer Science, New Stream Cipher Designs, The eSTREAM Finalists*, vol. 4986, pp. 244–266, 2008.

[5] J. Hoepman G. de Koning Gans and F. Garcia, "A practical attack on the mifare classic," in *Lecture Notes in Computer Science, vol 5189, Smart Card Research and Advanced Applications,(CARDIS 2008)*, pp. 267–288, Egham, United Kingdom, September 2008.

[6] I. Goldberg and D. Wagner, "Randomness and the netscape browser," *Dr. Dobb's Journal—Software Tools for the Professional Programmer*, vol. 21, no. 1, pp. 66–71, 1996.

[7] C. Meyer K. Michaelis and J. Schwenk, "Randomly failed! the state of randomness in current java implementations," in *Lecture Notes in Computer Science, vol 7779, Topics in Cryptology,(CT-RSA 2013)*, pp. 129–144, San Frnacisco, California, USA, February 2013.

[8] Z. El Abidine Guennoun K.Charif, A. Drissi, "A pseudo random number generator based on chaotic billiards," *International Journal of Network Security*, vol. 19, no. 3, pp. 479–486, 2017.

[9] H. Mahmood N. A. Saqib, M. Zia and M. A. Khan, "On generating high-quality random numbers," *Journal of Circuits, Systems and Computers*, vol. 26, no. 2, 2017.

[10] C. Paar and J. Pelzl, *Understanding Cryptography (1ed)*. London: Springer-Verlag Berlin Heidelberg, 2010.

[11] G. Chen Y. Tian and J. Li, "On the design of trivium," *IACR Cryptology ePrint Archive*, p. 431, 2009.

[12] J. Zaman and R. Ghosh, "A review study of nist statistical test suite: Development of an indigenous computer package," *arXiv preprint*, vol. 1208, no. 5740, 2012.

Biography

Latoya Jackson has a B.S. in Mathematics from Morgan State University, Baltimore, Maryland, USA and an M.S. in Applied Computer Science from Columbus State University in Columbus, Georgia, USA. She has special interest in cryptography, information security and algorithm analysis.

Yesem Kurt Peker is an Associate Professor of Computer Science at the TSYS School of Computer Science at Columbus State University in Columbus, Georgia. She holds B.Sc. degrees in Computer Engineering and Mathematics and Master's Degree in Mathematics from Middle East Technical University in Ankara, Turkey, and PhD in Mathematics from Indiana University Bloomington in the United States. Her research focuses on computer and network security, in particular cryptography. She also is interested in computer science and cybersecurity education for college students as well as younger generation.

Cryptanalysis and Improvement of a Smart Card Based Authentication Scheme for Multi-server Architecture Using ECC

Tao Wan[1], Xiaochang Liu[1], Weichuan Liao[2], and Nan Jiang[1]
(Corresponding author: Tao Wan)
School of Information Engineer, East China Jiaotong University, China[1]
Huangjiahu E Rd, Qingshanhu Qu, Nanchang Shi, Jiangxi Sheng 330029, China
School of Science, East China Jiaotong University, China[2]
(Email: wantao217@163.com)
(Received Apr. 4, 2018; Revised and Accepted Oct. 18, 2018; First Online June 16, 2019)

Abstract

Authentication and key agreement protocol becomes an important security issue for multi-server architecture. Wei *et al.* demonstrated that Pippal *et al.*'s protocol has several drawbacks and proposed an improved authentication scheme for multi-server architecture using smart card and password. They claimed that their scheme achieves intended security requirements and is more appropriate for practical applications. In this paper, we indicate that their scheme cannot resist user impersonation attack, cannot protect user's anonymity, unable to check user password in time and is also vulnerable to Denial of Service attack. To enhance the security of Wei *et al.*'s protocol, we propose a secure biometric-based authentication scheme for multi-server environment based on elliptic curve cryptography using smart card. Compared with other related schemes, the security analysis and performance evaluation show that our proposed scheme can provide stronger security.

Keywords: Authentication; Biometric-based; Key Agreement; Multi-Server; Smart Card

1 Introduction

With the rapid development of Internet applications, an increasing number of remote user authentication schemes are usually used to provide services to users. In the early, most authentication schemes are based on password. Unfortunately, as widely used in real-life settings, there were vulnerable to some attacks, such as dictionary attack and compromised stolen-verifier attack. To overcome these attacks, smart card based password authentication schemes [2, 3, 6, 10, 13, 15, 23, 31] have been proposed, which become one of the most general authentication scheme. However, most of these schemes based on the single-server, when users need to obtain different services from multiple servers, they not only have to register to different servers, but also need to remember a large number of identity and password. Obviously, it is very difficult and unsafe for users to remember and manage multiple information. In order to solve this problem, authentication schemes for the multi-server environment [5, 7, 11, 12, 18–21, 24–27, 29] have been proposed in recent year.

Recently, Lee *et al.* [16] analyzed Hsiang and Shih's scheme [9] and pointed out that their scheme is vulnerable to masquerade attack and server spoofing attack, and it cannot provide mutual authentication since the clerical error. To overcome the security flaws of Hsiang and Shih's scheme, Lee *et al.* proposed a secure dynamic ID based authentication scheme. But Li *et al.* [17] found that Lee *et al.*'s scheme is still vulnerable to forgery attack and server spoofing attack. Nevertheless, Chang *et al.* [4] indicated that their scheme is sensitive to the forgery attack. In 2013, He and Wu [8] demonstrated that Wang and Ma's scheme [28] is vulnerable to stolen smart card and leak of verifier attack and introduced the improvement scheme. Unfortunately, Pippal *et al.* [22] revealed that their scheme is still susceptible to impersonation attack, privileged insider attack and off-line password guessing attack. To solve above-mentioned security flaws, in 2014, Wei and Liu [30] proposed improvement of a robust smart card authentication scheme for multi-server architecture. But, we identify that Wei *et al.*'s scheme not only is vulnerable to DoS attack, user impersonation attack, but also lacks timely password check and users are easily tracked.

The remainder of this manuscript is organized as follows. We review the robust smart card authentication scheme for multi-server architecture proposed by Wei *et al.* in Section 2. We analyze the security flaws of Wei *et al.*'s scheme in Section 3. We present a proposed protocol in Section 4. We compare the performance of our proposed scheme with the previous schemes in Section 5. We

conclude this paper in Section 6.

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

Here we will review Wei *et al.*'s smart card based authentication scheme for multi-server architecture. The notations used throughout this paper are summarized as Table 1.

Their scheme involves three participants, the login user (U_i), the remote server (S_j) and the registration center (RC). Their scheme can be divided into four phase: initialization phase, registration phase, login and authentication phase and password change phase. We show the login and authentication phases in Figure 1. More details are provided in the following.

Table 1: Notations used in the paper

Symbols	Their meaning
RC	*the registration center*
U_i	*the i_{th} user*
UID_i	*the i_{th} user's identity*
S_j	*the j_{th} application server*
SID_j	*the j_{th} application server's identity*
PW_i	*the user U_i's password*
p and q	*two large prime numbers*
$h(\cdot)$	*a secure one-way hash function*
$\|$	*the concatenation operation*
$\oplus$	*exclusive-OR operation*
x	*random nonce generated by U_i*
y	*random nonce generated by S_j*
SK_{ij}	*section key shared between U_i and S_j*

2.1 Initialization Phase

Step I1: The registration center RC selects two large prime numbers p and q and computes $p = 2q + 1$.

Step I2: The registration center RC chooses a random nonce $g \in Z_p^*$, picks a random number $r_j \in Z_p^*$ as the private key of the remote server $S_j (1 \leq j \leq k)$, and sets $t = g^{\prod_{j=1}^{k} r_j} \bmod p$.

Step I3: The registration center RC selects a secure one-way hash function $h(\cdot) : \{0,1\}^* \rightarrow Z_p^*$.

2.2 Registration Phase

In Wei *et al.*'s scheme, the registration phase consists of two sub-phases, the server registration phase and the user registration. In this phase, the server and the user should register themselves to the registration center RC and obtains secret information to initial the system.

2.2.1 Server Registration Phase

This phase is executed between the application server S_j and the registration center RC. This registration phase consists of the following steps:

Step S1: The application server S_j sends a registration request along with its identity SID_j to the registration center RC, if he/she wishes to become a registered server.

Step S2: Receiving the registration request from the remote server S_j, the registration center RC assigns the value r_j to the remote server S_j.

Step S3: And then sends $\{r_j, t, p, q, h(\cdot)\}$ to the remote server S_j through a secure channel.

2.2.2 User Registration Phase

When a user wishes to access any services provided by the registered servers, he/she must first register himself/herself. This registration phase consists of the following steps:

Step U1: The user U_i freely chooses an identity UID_i, a private password PW_i and a random number b, then transmits the registration request information $\{UID_i, h(PW_i \| b)\}$ to the registration center RC via a secure channel.

Step U2: Upon getting the registration information from U_i, the registration center RC continues to compute $V_{ij} = h(t \| r_j \| UID_i)$, $S_{ij} = V_{ij} \oplus h(UID_i \| h(PW_i \| b))$ when UID_i is valid, otherwise rejects the user registration request.

Step U3: The registration center RC securely issues the smart card containing $\{(S_{i1}, S_{i2}, \ldots, S_{ik}), p, q, h(\cdot)\}$ to the user U_i.

Step U4: After receiving the issued smart card, the user U_i stores the random nonce b into the smart card.

2.3 Login and Authentication Phase

When a legal user U_i wants to access the resources provided by remote server S_j, he/she first attaches the smart card to a device reader, and inputs his/her identity UID_i and password PW_i. Then, as illustrated in Figure 1, the login and authentication mechanism is performed as follows:

Step V1: The smart card first computes

$$V_{ij} = S_{ij} \oplus h(UID_i \| h(PW_i \| b)),$$

then generates a random nonce x and computes

$$\begin{aligned} W_{ij} &= h(UID_i || SID_j)^x \bmod p, \\ W_{ij}^* &= W_{ij} \oplus V_{ij}, \\ R_1 &= h(UID_i || W_{ij}^* || T_i). \end{aligned}$$

The smart card sends the login request message $M_1 = \{UID_i, W_{ij}^*, R_1, T_i\}$ to the remote server S_j.

Step V2: Upon receiving the message from the user U_i, the remote server S_j checks whether UID_i is valid and $T_i' - T_i$ is less than $\triangle T$. Moreover, S_j verifies whether $R_1' = h(UID_i||W_{ij}^*||T_i)$ is equal to R_1. If not, the communication is simply terminated.

Step V3: The remote server S_j chooses a random number y, and first computes

$$\begin{aligned} B_{ij} &= h(UID_i||SID_j)^y \bmod p, \\ V_{ij}' &= h(t||r_j||UID_i), \\ W_{ij}' &= W_{ij}^* \oplus V_{ij}', \\ Z_{ij} &= (W_{ij}')^y \bmod p, \\ R_2 &= h(UID_i||W_{ij}'||B_{ij}||Z_{ij}||T_j). \end{aligned}$$

Furthermore, the remote server S_j sends the response message $M_2 = \{B_{ij}, R_2, T_j\}$ to user U_i.

Step V4: After getting the message M_2, the smart card checks whether $T_i' - T_i \leq \triangle T$, if T_j is valid, the smart card computes $Z_{ij}' = B_{ij}^x \, mod \, p$, and checks whether $R_2' = h(UID_i\|W_{ij}\|B_{ij}\|Z_{ij}'\|T_j)$ is equal to R_2. If not, the smart card terminates the communication.

Step V5: The smart card computes

$$\begin{aligned} SK_{ij} &= h(UID_i\|W_{ij}\|B_{ij}\|Z_{ij}'), \\ R_3 &= h(UID_i||W_{ij}^*||B_{ij}\|Z_{ij}'\|T_k). \end{aligned}$$

Then, smart card transmits the message $M_3 = \{UID_i, R_3, T_k\}$ to the remote server S_j.

Step V6: Upon getting the message M_3, the S_j checks UID_i and T_k. If they are both valid, S_j checks $R_3'? = h(UID_i\|W_{ij}^*\|B_{ij}\|Z_{ij}\|T_k)$. If not, the server S_j terminates the communication. Otherwise, S_j generates the session key

$$SK_{ij}' = h(UID_i\|W_{ij}'\|B_{ij}\|Z_{ij}).$$

2.4 Password Change Phase

This phase is invoked whenever U_i wants to change his password PW_i to a new password PW_i^{new} .

Step P1: U_i inserts his smart card and inputs his identity UID_i and password PW_i.

Step P2: For each $(1 \leq j \leq k)$, the smart card computes $S_{ij}^{new} = S_{ij} \oplus h(UID_i||h(PW_i||b)) \oplus h(UID_i||h(PW_i^{new}||b))$.

Step P3: The smart card replaces $(S_{i1}, S_{i2}, \ldots, S_{ik})$ with $(S_{i1}^{new}, S_{i2}^{new}, \ldots, S_{ik}^{new})$.

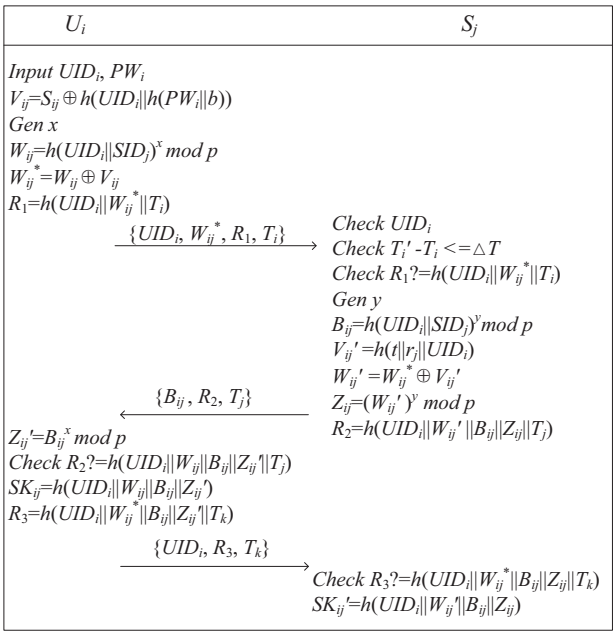


Figure 1: Login and authentication phase of Wei *et al.*'s scheme

3 Security Analysis of Wei *et al.*'s Scheme

In Wei *et al.*'s scheme, they proposed an improved smart card authentication scheme for multi-server architecture that can resist various well-known attacks, such as off-line password guessing attacks, impersonation attacks and privileged insider attacks. Unfortunately, we find that their scheme still has many vulnerabilities. an attacker can launch denial of service attack, because the user transmits data to remote server through the public channel. Secondly, an adversary can initiate impersonation attack once the stolen. Besides, there is no password checking after the user inputs his/her password, the wrong password cannot be found in time. Moreover, the user's behavior is easily to be traced. The detailed description is as follows.

3.1 Denial of Service Attack

From the login and authentication phase of Wei *et al.*'s scheme, we find that any attacker Z can easily forge a login request message that can pass S_j's authentication by eavesdropping a valid login request message and then launch DoS attack on the server.

An malicious attacker Z may eavesdrop the valid login request message $\{UID_i, W_{ij}^*, R_1, T_i\}$ that the user U_i transmitted to the server S_j and compute $R_1' = h(UID_i||W_{ij}^*||T_i')$, where T_i' is the current time. Then Z can forge the request message $\{UID_i, W_{ij}^*, R_1', T_i'\}$ that can pass S_j's verification.

After that, the server S_j select a random y, and com-

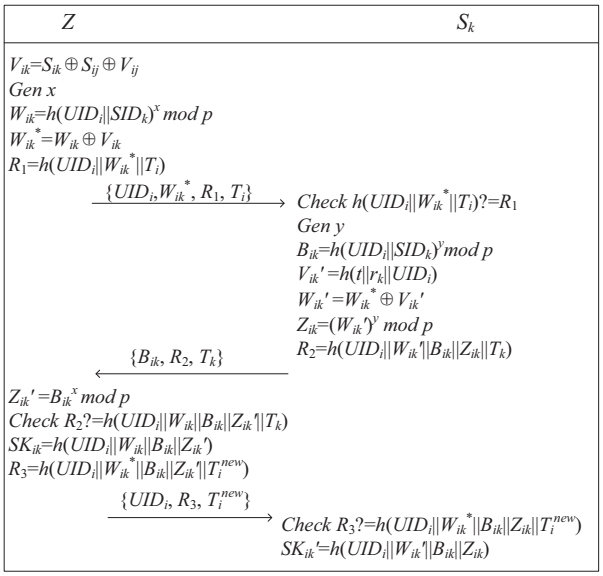


Figure 2: User impersonation attack on Wei *et al.*'s Scheme

putes

$$\begin{aligned} B_{ij} &= h(UID_i||SID_j)^y \bmod p, \\ V_{ij}' &= h(t||r_j||UID_i), \\ W_{ij}' &= W_{ij}^* \oplus V_{ij}', \\ Z_{ij} &= (W_{ij}')^y \bmod p, \\ R_2 &= h(UID_i||W_{ij}'||B_{ij}||Z_{ij}||T_j), \end{aligned}$$

where T_j is the current timestamp.

Then, S_j transmits message $M_2 = \{B_{ij}, R_2, T_j\}$ to the user U_i. The attacker Z will intercept the message to terminate the communication.

By this way, any attacker can launch DoS attack on the server S_j which will cause the computing and communication loss of S_j.

3.2 User Impersonation Attack

As shown in Wei *et al.*'s scheme, any registered server S_j can compute $V_{ij} = h(t||r_k||UID_i)$. Under the condition that the server S_j was captured by an attacker Z, Z can impersonate as U_i to log in to any registered server (*e.g.*, S_k) by stealing U_i's smart card without knowing UID_i and PW_i as show in Figure 2. The procedure is as follow:

- The attacker Z retrieves S_{ij} and S_{ik} from U_i's smart card, then computes $V_{ik} = S_{ik} \oplus S_{ij} \oplus V_{ij}$;
- Z generates a random number x, and computes

$$\begin{aligned} W_{ik} &= h(UID_i||SID_k)^x \bmod p, \\ W_{ik}^* &= W_{ik} \oplus V_{ik}, \\ R_1 &= h(UID_i||W_{ik}^*||T_i). \end{aligned}$$

  Then, Z forwards $M_1 = \{UID_i, W_{ik}^*, R_1, T_i\}$ to S_k;
- Upon receiving the login request message M_1, the remote server S_k checks the validity of T_i and compares $h(UID_i||W_{ik}^*||T_i)$ with R_1. Because they are equivalent, S_k will accept the login request;
- The server S_k generates a random number y to compute

$$\begin{aligned} B_{ik} &= h(UID_i||SID_k)^y \bmod p, \\ V_{ik}' &= h(t||r_k||UID_i), \\ W_{ik}' &= W_{ik}^* \oplus V_{ik}', \\ Z_{ik} &= (W_{ik}')^y \bmod p, \\ R_2 &= h(UID_i||W_{ik}'||B_{ik}||Z_{ik}||T_k). \end{aligned}$$

  Then, S_k transmits $M_2 = \{B_{ik}, R_2, T_k\}$ to U_i;
- Z intercepts M_2, and computes $Z_{ik}' = B_{ik}^x \bmod p$, checks whether $h(UID_i||W_{ik}||\ B_{ik}||Z_{ik}'||T_k)$ is equal to R_2. If it is holds, Z computes

$$\begin{aligned} SK_{ik} &= h(UID_i||W_{ik}||B_{ik}||Z_{ik}'), \\ R_3 &= h(UID_i||W_{ik}^*||B_{ik}||Z_{ik}'||T_i^{new}). \end{aligned}$$

  Finally, Z sends $M_3 = \{UID_i, R_3, T_i^{new}\}$ to S_k;
- After receiving M_3, S_k checks the validity of T_i^{new} and verifies whether $h(UID_i||W_{ik}^*\ ||B_{ik}||Z_{ik}||T_i^{new})$ is equal to R_3. If it holds, S_k generates the session key $SK_{ik}' = h(UID_i||W_{ik}'||B_{ik}||Z_{ik})$. Obviously, $SK_{ik}' = SK_{ik}$, a shared session key is established between the attacker Z and the remote server S_k.

At last, the attacker Z logs in to the server S_k by masquerading as U_i. Therefore, Wei *et al.*'s scheme cannot withstand user impersonation attack.

3.3 Unable to Check Password in Time

In the login and authentication phase of Wei *et al.*'s scheme, the device reader cannot check the identity UID_i and password PW_i of U_i in time, which may consume the computational and communication cost of remote server and smart card. The detailed description is as follows.

Once a legal user U_i attaches his/her smart card to a device reader, inputs his/her identity UID_i and an error password PW_i'. The smart card computes $V_{ij} = S_{ij} \oplus h(UID_i||h(PW_i'||b))$, and selects a random number x to computes

$$\begin{aligned} W_{ij} &= h(UID_i||SID_j)^x \bmod p, \\ W_{ij}^* &= W_{ij} \oplus V_{ij}, \\ R_1 &= h(UID_i||W_{ij}^*||T_i). \end{aligned}$$

Afterwards, the smart card transmits the message $M_1 = \{UID_i, W_{ij}^*, R_1, T_i\}$ to a remote server S_j.

After receiving the message M_1, S_j checks the validity of T_i and whether $h(UID_i||W_{ij}^*||T_i)$ is equal to R_1. Obviously, it holds. Then S_j chooses a random number y,

and computes

$$\begin{aligned} B_{ij} &= h(UID_i||SID_j)^y \, mod \, p, \\ V'_{ij} &= h(t||r_j||UID_i), \\ W'_{ij} &= W^*_{ij} \oplus V'_{ij}, \\ Z_{ij} &= (W'_{ij})^y \, mod \, p, \\ R_2 &= h(UID_i||W'_{ij}||B_{ij}||Z_{ij}||T_j). \end{aligned}$$

Eventually, the remote server S_j transfer the response message $M_2 = \{B_{ij}, R_2, T_j\}$ to U_i.

Upon getting the message M_2, the smart card computes $Z'_{ij} = B^x_{ij} \, mod \, p$, and check whether $R'_2 = h(UID_i||W_{ij}||B_{ij}||Z'_{ij}||T_j)$ is equal to R_2. Since U_i input an error password PW'_i, $V_{ij} = S_{ij} \oplus h(UID_i||h(PW'_i||b))$ will not equal to $V'_{ij} = h(t||r_j||\ UID_i)$, the smart card terminates the communication.

From the above discussion, we know that the error password was not be found in time, smart card and remote server have waste a large number of computational and communication resource.

3.4 No Provision of User Anonymity

With the wide application of network technology, the protection of user's privacy have received more and more attentions, user anonymity is a desirable property for remote user authentication. In Wei *et al.*'s protocol, the identity UID_i of user U_i is static, which will cause the user's login request be traced.

4 The Proposed Protocol

Based on the cryptanalysis of Wei *et al.*'s scheme, we present an enhanced biometric based authentication with key-agreement protocol for multi-server architecture based on elliptic curve cryptography. The proposed protocol consists of four phases: initialization phase, registration phase, login and authentication phase, and password change phase. There are also three participants: the user U_i, remote server S_j and registration center RC.

4.1 Initialization Phase

Registration server RC generates following parameters in order to initialize the system.

Step I1: The registration center RC chooses an elliptic curve equation E with an order n.

Step I2: The registration center RC selects a base point Q over E and chooses a one-way cryptographic hash function $h(\cdot)$.

Step I3: The registration center RC publishes the information $\{E, Q, h(\cdot)\}$.

4.2 Registration Phase

In our proposed protocol, the registration phase consists of two sub-phases, namely, server registration phase and user registration phase. In this phase, the server S_j and the user U_i should register themselves to the registration center RC and obtains secret information to initial system.

4.2.1 Server Registration Phase

In this phase, the remote server S_j sends a registration request to the registration center RC in order to become an authorized server. The registration process according to the following steps:

Step S1: The remote server S_j computes public key $P_b = P_r \cdot Q$ and sends registration request $\{P_b, SID_j\}$to RC.

Step S2: The registration center RC sends PSK to the remote server S_j, which can be used in further phases of authentication.

4.2.2 User Registration Phase

When a user wants to access the services of registered servers, he/she must register himself/herself, as shown in Figure 3. This registration process according to the following steps.

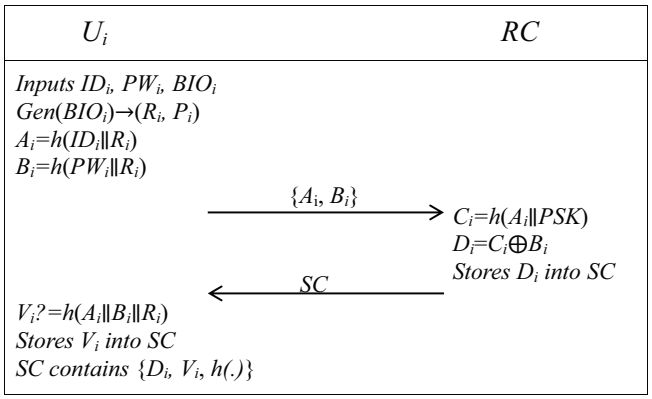


Figure 3: User registration phase of the proposed protocol

Step U1: The user U_i chooses an identity ID_i, password PW_i. Then the user U_i imprints his personal biometric information BIO_i at a sensor. The sensor sketches BIO_i to extract an unpredictable binary string R_i and an auxiliary binary string P_i from $Gen(BIO_i) \rightarrow (R_i, P_i)$. Then, sensor stores P_i in the memory. Next the user U_i computes $A_i = h(ID_i||R_i)$, $B_i = h(PW_i||R_i)$. Finally, the user U_i sends a request message $\{A_i, B_i\}$ to RC via a secure channel.

Step U2: Upon receiving the request message, RC computes $C_i = h(A_i||PSK), D_i = B_i \oplus C_i$.

Step U3: RC stores the parameters $\{D_i,\ h(\cdot)\}$ into a new smart card and delivers it to the user U_i via a secure channel.

Step U4: Upon getting the message, the user U_i computes $V_i = h(A_i||B_i||R_i)$ and stores $\{V_i\}$ into smart card. Thus the smart card finally contains the parameters $\{D_i, V_i, h(.)\}$.

4.3 Login and Authentication Phase

When a legal user U_i wants to login into some remote server S_j, he/she first attaches the smart card to a device reader, and inputs ID_i and PW_i. Next, the user U_i imprints his biometric information BIO_i at a sensor. After that, sensor sketches user U_i's biometric information BIO_i and recovers the string R_i from $Rep(BIO_i, P_i) \rightarrow R_i$. Then, the concrete login and authentication procedure, as shown in Figure 4, the login and authentication mechanism is performed as follows:

Step V1: The smart card SC computes $A_i = h(ID_i||R_i)$, $B_i = h(PW_i||R_i)$, and then verifies whether V_i is equal to $h(A_i||B_i||R_i)$. If V_i is invalid, SC terminates the communication; otherwise, the smart card SC generates a random number x and calculates $K = x \cdot Q$, $K' = x \cdot P_b$, $AID_i = A_i \oplus K'$, $C_i = D_i \oplus B_i$ and $M_1 = h(AID_i||C_i||K||K'||T_i)$. Then the smart card SC sends the login request message $\{M_1, K, AID_i, T_i\}$ to the remote server S_j.

Step V2: Upon receiving the message from the user U_i, the remote server S_j checks whether T_i' - T_i is less than $\triangle T$. The remote server computes $K' = P_r \cdot K$, $A_i = AID_i \oplus K'$, $C_i = h(A_i||PSK)$ and verifies whether M_1 is equal to $h(AID_i||C_i||K||K'||T_i)$. If the condition holds, the remote server S_j authenticates the user U_i, otherwise the process can be terminated.

Step V3: The remote server S_j further generates a random number N_1 and computes $M_2 = A_i \oplus N_1, M_3 = h(A_i||K'||SID_j||N_1)$ and $SK_{ij} = h(A_i||K'||C_i||SID_j||N_1)$. Furthermore, the remote server S_j sends the response message $\{M_2, M_3\}$to the user U_i.

Step V4: After getting the message M_2 and M_3, the user U_i computes $N_1 = M_2 \oplus A_i$ and verifies whether M_3 is equal to $h(A_i||K'||SID_j||N_1)$. If the condition holds, the user U_i authenticates the remote server S_j, otherwise the process can be terminated. Then, the user computes $SK_{ij} = h(A_i||K'||C_i||SID_j||N_1)$, $M_4 = h(SK_{ij}||K'||N_1)$ and sends the message $\{M_4\}$ to the remote server S_j.

Step V5: Upon receiving the message, the remote server S_j verifies whether M_4 is equal to $h(SK_{ij}||K'||N_1)$ and reconfirms the authenticity of U_i. Now, the user U_i and the server S_j can start communication with the computed session key SK_{ij}.

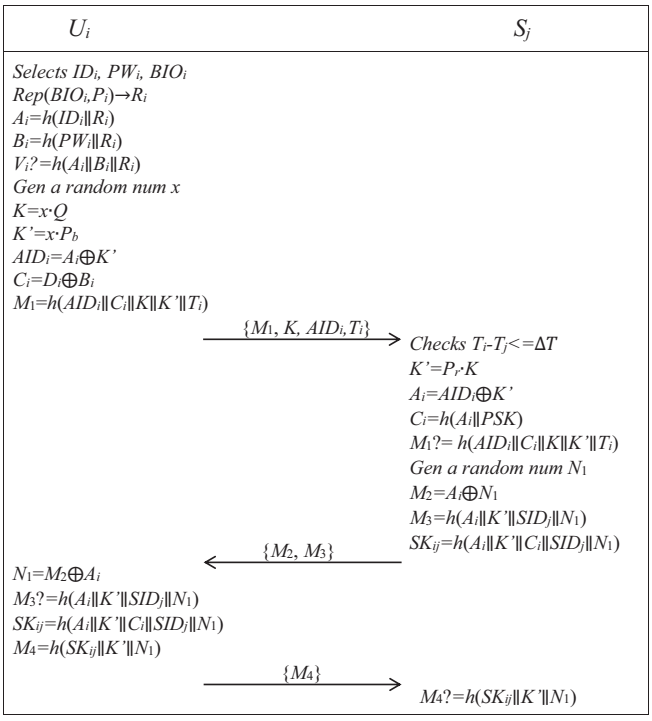


Figure 4: User registration phase of the proposed protocol

4.4 Password Changing Phase

This procedure invokes when a user (U_i) wish to update his/her existing password with new one. In this procedure, the user U_i can change his/her password as follows:

Step P1: The user U_i inserts smart card SC and inputs ID_i, PW_i and BIO_i.

Step P2: The smart card SC computes $A_i = h(ID_i||R_i)$, $B_i = h(PW_i||R_i)$, and then verifies the condition whether V_i is equal to $h(A_i||B_i||R_i)$. If this verification is valid, the smart card SC asks the user U_i for a new password. Otherwise, password change phase is terminated immediately by the smart card SC.

Step P3: The user U_i chooses a new password PW_i^{new} and then computes $A_i^{new} = h(ID_i||R_i)$, $B_i^{new} = h(PW_i||R_i)$, $C_i^{new} = h(A_i||PSK)$, $D_i^{new} = B_i \oplus C_i$, $V_i^{new} = h(A_i||B_i||R_i)$.

Step P4: In the memory, smart card SC respectively replaces D_i with D_i^{new} and V_i with V_i^{new}.

5 Analysis of the Proposed Protocol

In a multi-server architecture, there are three requirements for an authentication and key agreement protocol, namely, security, functionality and efficiency. In this section, we first present security analysis of our proposal, and then examine its performance in terms of functionality and efficiency by comparing it with previous related works.

5.1 User Anonymity

In our protocol, the real identity of user is not revealed throughout all the phases of communication. In the user registration phase, U_i submits $A_i = h(ID_i||R_i)$ and the real identity is protected with a one-way hash function. During the login phase, the parameter A_i is converted as anonymous in the form of $A_i = AID_i \oplus K'$. The identity is dynamic for every login session, due to its association with a random number x, where $K = x \cdot Q$ and $K' = x \cdot P_b$. An adversary cannot retrieve the x in anyway. Moreover, it is believed to be impossible to compute K' from K and P_b because of *ECDLP*. In the other hand, our protocol achieves the user untraceability. In the user login phase, the user U_i sends the message $\{M_1, K, AID_i, T_i\}$ to the remote server S_j. All the parameters are dynamic and dose not disclose the identity of U_i. Hence, our protocol achieves user anonymity and untraceability.

5.2 Resistance to Denial-of-Service Attack

The Denial-of-Service attack diminishes or eliminates the server's expected capability to make the server unavailable. With the help of timestamp T_i, the remote server S_j checks the freshness and legality of $M_1 = h(AID_i||C_i||K||K'||T_i)$ in the login request message. The current timestamp does not match the previous M_1 which is sent by adversary. Moreover, our scheme applies the fuzzy extractor to satisfy the usage requirements of biometrics. As a result, our scheme is secure against the Denial-of-Service attack.

5.3 Resistance to User Impersonation Attack

Under the user impersonation attack, an adversary who is an outsider hackerS tries to impersonate user U_i without the password PW_i or biometric information BIO_i. If an adversary wants to masquerade a legitimate user U_i, he/she requires to build a login message $\{M_1, K, AID_i, T_i\}$, where $M_1 = h(AID_i||C_i||K||K'||T_i)$, $K = x \cdot Q$, $AID_i = Ai \oplus K'$. Conversely, the adversary can barely compute $K' = x' \cdot Q$ and $K'' = x' \cdot P_b$ by choosing his/her own random number x'. But the adversary can't compute rest of the two parameters, due to the unavailability of valid ID_i, PW_i and R_i. Hence, our protocol is secure against the user impersonation.

5.4 Resistance to Server Impersonation Attack

Our protocol protects the server impersonation attack and it's description is given below:

- In order to act as a legitimate server, the adversary eavesdrops the valid login request message $\{M_1, K, AID_i, T_i\}$ that the user U_i transmitted to the server S_j, and generates a random number P'_r and N'_1. Then S_j computes $K'' = P'_r \cdot K$, $A'_i = AID_i \oplus K''$, $C'_i = h(A'_i||PSK)$, $M'_2 = A'_i \oplus N'_1$, $M'_3 = h(A'_i||K''||SID_j||N'_1)$, $SK'_{ij} = h(A'_i||K''||C'_i||SID_j||N'_1)$. The adversary sends $\{M'_2, M'_3\}$ to U_i.
- Upon receiving the $\{M'_2, M'_3\}$, the user U_i computes $N'_1 = A_i \oplus M_2$ and $M_3 = h(A_i||K'||SID_j||N'_1)$. Here, U_i identities it as a fake response from the malicious server because of M_3 is not equal to M'_3 and terminates the session. Hence, our protocol can resist the server impersonation attack.

5.5 Resistance to Smart Card Stolen Attack

The adversary can extract the information $\{D_i, V_i, h(\cdot)\}$ stored in the smart card by means of power analysis. Assume a legal user's smart card is stolen by an adversary and extracted the information $\{D_i, V_i, h(\cdot)\}$. Then, the adversary may try to get ID_i, PWi and R_i from the extracted information. However, adversary cannot obtain any valuable information from these values, where $D_i = B_i \oplus C_i$, $C_i = h(A_i||PSK)$, $B_i = h(PW_i||R_i)$, $A_i = h(ID_i||R_i)$ and $V_i = h(A_i||B_i||R_i)$ since all the important parameters such as ID_i, PW_i and R_i are protected by a one-way hash function. The adversary cannot obtain any login information using the smart card stored parameters D_i and V_i. At the same time, guessing the real identity ID_i, password PW_i and biometric R_i is impractical. Therefore, our protocol is secure against smart card stolen attack.

5.6 Resistance to Replay Attack

If an adversary intercepts the communication message $\{M_1, K, AID_i, T_i\}$ between U_i and S_j, he tries to replay them to S_j to masquerade as a legal user. However, once the message is replayed, the server S_j can immediately detect the attack and reject the request due to the apply of timestamp T_i. Hence, our protocol is secure against replay attack.

5.7 Resistance to Privileged Insider Attack

During our protocol, U_i does not send his ID_i, password PW_i or his biometrics BIO_i in user registration phase. U_i submits only $A_i = h(ID_i||R_i)$, $B_i = h(PW_i||R_i)$ to RC instead of original credentials. Hence, an insider cannot obtain the original sensitive information of any user. On the other hand, the $M_1 = h(AID_i||C_i||K||K'||T_i)$ is invalid in which P_r is unobtainable. Therefore, our protocol resists to privileged insider attack.

5.8 Resistance to Password Guessing Attack

An adversary may try to guess the password PW_i from the extracted smart card stored parameters $\{D_i, V_i, h(\cdot)\}$.

The stored parameter contains the password PW_i in the form $B_i = h(PW_i||R_i)$ where $Gen(BIO_i) \rightarrow (R_i, P_i)$. An adversary attempts to verify the condition $V_i? = h(A_i||B_i||R_i)$ while constantly guessing PW_i. Adversary needs the value of ID_i and R_i of U_i in order to achieve the password guessing attack. However, the value of R_i is nowhere stored and an adversary cannot get the value of ID_i. As a result, the adversary cannot guess the correct password PW_i. Therefore, our protocol resist to password guessing attack.

5.9 Forward Secrecy

Perfect forward secrecy protects the session keys even if long-term key is retrieved. Specifically, the session key in the proposed scheme is generated as $SK_{ij} = h(A_i||K'||C_i||SID_j||N_1)$ and the long term private key of the server PSK in $C_i = h(A_i||PSK)$ is shielded with a hash function and is not possible to derive due to its one-way property. Although the long term key is compromised with an adversary, he/she still cannot compute a valid session key, the parameter $K' = P_r \cdot K$ and $K = x \cdot Q$ is dynamic due to its association with random generated number x, which is not possible to extract due to the reason of *ECDLP*. Therefore, our protocol provides perfect forward secrecy.

5.10 Performance and Functionality Comparisons

In this section, we compare our proposed protocol with several related schemes [8, 22, 28, 30]. In Table 2, we provide the comparison based on the key security of these schemes, while we compare their efficiency in terms of computation and communication cost . The computation cost of the protocol is the times of executing operations. The following notations are used in Table 2.

- T_e: modular exponentiation operation;
- T_m: modular multiplication/inverse operation;
- T_h: hash operation;
- T_{epm}: the time for executing a scalar multiplication operation of elliptic curve.

We also define i as the length of one parameter in the transmitted messages, such as the length of R is i and the length of $\{R, CID_i\}$ is $2i$. As Amin and Islam [1] executed various cryptographic operations using MIRACL C/C++ Library, the computation cost for T_h is approximately 0.0004ms, T_e is approximately 1.8269ms and T_m is approximately 0.0147ms. As per Kilinc and Yanik [14] experiment on a personal computer involving a processor with Dual CPU E2200 2.20 GHz along with RAM size of 2048MB, the computation cost for T_{epm} is approximately 2.229ms.

In Table 2, we summarize the efficiency comparison according to the computation cost and communication cost between our protocol and other schemes [8, 22, 28, 30] in case that the login and authentication phase is done. From Table 4, it is easy to see that our scheme is more efficient than Wei *et al.*'s scheme [30], He *et al.*'s scheme [8] and Wang *et al.*'s scheme [28]. Moreover, our proposed protocol is lower computation cost than those of Wei *et al.*'s scheme [30].

From Table 3, it can be observed that the proposed protocol is more secure than the other four schemes. Our new protocol satisfies all the security requirements listed in Table 3. Wei *et al.*'s scheme [30] only satisfy five of the nine requirements, respectively. Pippal *et al.*'s scheme [22], He *et al.*'s scheme [8] and Wang *et al.*'s scheme [28] only satisfies three of the nine requirements. Hence, our scheme achieves stronger security than their solutions.

6 Conclusions

In this paper, we analyzed Wei *et al.*'s smart card based multi-server authentication scheme. Our analysis reveals its inherent security vulnerabilities, *i.e.*, denial of service attack, impersonation attack, unable to check password in time and no provision of user anonymity. In addition, this paper proposed an enhanced biometric based authentication with key agreement protocol for multi-server architecture based on elliptic curve cryptography. The mutual authentication of the proposed protocol achieved significant features such as biometric authentication, elliptic curve cryptography, with less computational and communication cost. Furthermore, the comparison results evidently indicate that our protocol is more secure than other schemes. Thus, our protocol is more feasible for practical applications.

Acknowledgments

This research was supported by the National Natural Science Foundation of China (No.61962022), Key Research and Development Plan of Jiangxi Province (No.20192BBE50077), the project of Education Department of Jiangxi Province (No. GJJ160510).

References

[1] R. Amin, S. H. Islam, and M. K. Khan, "A two-factor rsa-based robust authentication system for multiserver environments," *Security & Communication Networks*, vol. 2017, no. 13, pp. 1–15, 2017.

[2] R. Amin, S. K. Islam, and G. P. Biswas, "An efficient and practical smart card based anonymity preserving user authentication scheme for tmis using elliptic curve cryptography," *Journal of Medical Systems*, vol. 39, no. 11, p. 180, 2015.

[3] T. Cao and S. Huang, "Cryptanalysis of a sensor smart card based password authentication scheme

Table 2: Efficiency comparison

	Wang *et al.*[19]	He *et al.*[5]	Pippal *et al.*[11]	Wei *et al.*[21]	Ours
User	$6T_h+3T_e$	$7T_h+2T_{epm}$	$4T_h+3T_e+T_m$	$7T_h+2T_e$	$7T_h+2T_{epm}$
Server	$6T_h+2T_e$	$6T_h+T_{epm}$	$3T_h+4T_e+T_m$	$6T_h+2T_e$	$4T_h+T_{epm}$
RC	$3T_h$	$3T_h$	$3T_h$	$3T_h$	T_h
Total	9.1405ms	6.6934ms	3.6872ms	7.3140ms	6.6922ms
Communication cost	6i	6i	5i	10i	7i

Table 3: Security comparison

	Wang *et al.*[19]	He *et al.*[5]	Pippal *et al.*[11]	Wei *et al.*[21]	Ours
User anonymity	No	No	No	No	Yes
Denial-of-Service attack	No	No	No	No	Yes
User impersonation attack	No	Yes	Yes	No	Yes
Server impersonation attack	Yes	No	Yes	Yes	Yes
Smart card stolen attack	No	No	No	No	Yes
Replay attack	Yes	Yes	Yes	Yes	Yes
Privileged insider attack	No	No	No	Yes	Yes
Password guessing attack	Yes	No	No	Yes	Yes
Forward secrcy	Yes	Yes	No	Yes	Yes

with user anonymity," *Sensor Letters*, vol. 11, no. 11, pp. 2149–2151(3), 2013.

[4] C. C. Chang, T. F. Cheng, and W. Y. Hsueh, "A robust and efficient dynamic identity-based multi-server authentication scheme using smart cards," *International Journal of Communication Systems*, vol. 29, no. 2, pp. 290–306, 2016.

[5] T. Y. Chen, M. S. Hwang, C. C. Lee, J. K. Jan, "Cryptanalysis of a secure dynamic ID based remote user authentication scheme for multi-server environment," in *Fourth International Conference on Innovative Computing, Information and Control (ICICIC'09)*, pp. 725–728, IEEE, 2009.

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

[7] T. H. Feng, C. H. Ling, and M. S. Hwang, "Cryptanalysis of tan's improvement on a password authentication scheme for multi-server environments," *International Journal of Network Security*, vol. 16, no. 4, pp. 318–321, 2014.

[8] D. B. He and S. H. Wu, "Security flaws in a smart card based authentication scheme for multi-server environment," *Wireless Personal Communications*, vol. 70, no. 1, pp. 323–329, 2013.

[9] H. C. Hsiang and W. K. Shih, "Improvement of the secure dynamic ID based remote user authentication scheme for multi-server environment," *Computer Standards & Interfaces*, vol. 31, no. 6, pp. 1118–1123, 2009.

[10] M. S. Hwang, Li-Hua Li, "A New Remote User Authentication Scheme Using Smart Cards", *IEEE Transactions on Consumer Electronics*, vol. 46, no. 1, pp. 28–30, Feb. 2000.

[11] M. S. Hwang, J. W. Lo, C. Y. Liu, S. C. Lin, "Cryptanalysis of a user friendly remote authentication scheme with smart card," *Pakistan Journal of Applied Sciences*, vol. 5, no. 1, pp. 99–100, 2005.

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

[13] Q. Jiang, J. F. Ma, G. S. Li, and X. H. Li, "Improvement of robust smart-card-based password authentication scheme," *International Journal of Communication Systems*, vol. 28, no. 2, pp. 383–393, 2015.

[14] H. H. Kilinc and T. Yanik, "A survey of sip authentication and key agreement schemes," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 2, pp. 1005–1023, 2014.

[15] M. Kumar, M. K. Gupta, and S. Kumari, "An improved efficient remote password authentication scheme with smart card over insecure networks,"

International Journal of Network Security, vol. 13, no. 3, pp. 167–177, 2011.

[16] C. C. Lee, T. H. Lin, and R. X. Chang, "A secure dynamic id based remote user authentication scheme for multi-server environment using smart cards," *Expert Systems with Applicationsn*, vol. 38, no. 11, pp. 13863–13870, 2011.

[17] X. Li, J. Ma, W. D. Wang, and J. S. Zhang, "A novel smart card and dynamic id based remote user authentication scheme for multi-server environments," *Mathematical & Computer Modelling*, vol. 58, no. 1-2, pp. 85–95, 2013.

[18] C. T. Li, M. S. Hwang, "An efficient biometrics-based remote user authentication scheme using smart cards", *Journal of Network and Computer Applications*, vol. 33, no. 1, pp. 1-5, 2010.

[19] C. T. Li, M. S. Hwang, "An online biometrics-based secret sharing scheme for multiparty cryptosystem using smart cards", *International Journal of Innovative Computing, Information and Control*, vol. 6, no. 5, pp. 2181–2188, 2010.

[20] C. H. Ling, W. Y. Chao, S. M. Chen, M. S. Hwang, "Cryptanalysis of dynamic identity based on a remote user authentication scheme for a multi-server environment," in *International Conference on Advances in Mechanical Engineering and Industrial Informatics (AMEII'15)* , pp. 981–986, 2015.

[21] H. T. Pan, C. S. Pan, S. C. Tsaur, M. S. Hwang, "Cryptanalysis of efficient dynamic ID based remote user authentication scheme in multi-server environment using smart card," in *12th International Conference on Computational Intelligence and Security*, pp. 590–593, 2017.

[22] R. S. Pippal, C. D. Jaidhar, and S. Tapaswi, "Robust smart card authentication scheme for multi-server architecture," *Wireless Personal Communications*, vol. 72, no. 1, pp. 729–745, 2013.

[23] R. Ramasamy and A. P. Muniyandi, "An efficient password authentication scheme for smart card," *International Journal of Network Security*, vol. 14, no. 3, pp. 180–186, 2012.

[24] T. Maitra, S. H. Islam, and R. Amin, "An enhanced multi-server authentication protocol using password and smart-card: cryptanalysis and design," *Security & Communication Networks*, vol. 3, no. 17, pp. 4615–4638, 2016.

[25] J. L. Tsai, "Efficient multi-server authentication scheme based on one-way hash function without verification table," *Computers & Security*, vol. 27, no. 3-4, pp. 115–121, 2008.

[26] T. Wan, N. Jiang, and J. F. Ma, "Cryptanalysis of two dynamic identity based authentication schemes for multi-server architecturet," *China Communications*, vol. 11, no. 11, pp. 125–134, 2014.

[27] T. Wan, N. Jiang, and J. F. Ma, "Cryptanalysis of a biometric-based multi-server authentication scheme," *International Journal of Security and its Application*, vol. 10, no. 2, pp. 163–170, 2016.

[28] B. Wang and M. D. Ma, "A smart card based efficient and secured multi-server authentication scheme," *Wireless Personal Communications*, vol. 68, no. 2, pp. 361–378, 2013.

[29] R. C. Wang, W. S. Juang, and C. L. Lei, "User authentication scheme with privacy-preservation for multi-server environment," *IEEE Communications Letter*, vol. 13, no. 2, pp. 157–159, 2009.

[30] J. H. Wei, W. F. Liu, and X. X. Hu, "Cryptanalysis and improvement of a robust smart card authentication scheme for multi-server architecture," *Wireless Personal Communications*, vol. 77, no. 3, pp. 2255–2269, 2014.

[31] H. Wijayanto and M. S. Hwang, "Improvement on timestamp-based user authentication scheme with smart card lost attack resistance," *International Journal of Network Security*, vol. 17, no. 2, pp. 160–164, 2015.

Biography

Tao Wan received her B.S. degree in Mathematics from Hunan University, Changsha, China, and received her M.S. and Ph.D. degree in Computer Science from Xidian University, Xi'an, China. She is now an associate professor at East China Jiaotong University. Her research interests include cryptography, network and information security, e-commerce security technology.

Xiaochang Liu received her B.S. degree in Software Engineering from North University of China, Taiyuan, China. She is currently a M.S. candidate at East China Jiaotong University, Nanchang, China. Her research interests include network and information security, e-commerce security technology.

Weichuan Liao received his B.S. and M.S. degree in Mathematics from Hunan University, Changsha, China. He is now an associate professor at East China Jiaotong University. His research interests include cryptography, network and information security.

Nan Jiang received his Ph.D. degree in Computer Application Technology from Nanjing University of Aeronautics and Astronautics, Nanjing, China. Now he is an associate professor at East China Jiaotong University. From 2013 to 2014 he is a research scholar in Complex Networks and Security Research Lab at Virginia Tech. His research interests include wireless sensor networks, wireless protocol and architecture, distributed computing and complex network theory.

Research on Cloud Service Security Measurement Based on Information Entropy

Tilei Gao[1,2], Tong Li[3], Rong Jiang[2], Ming Yang[2], and Rui Zhu[1]
(Corresponding author: Ming Yang)

School of Software, Yunnan University, Kunming 650091, China[1]
School of Information, Yunnan University of Finance and Economics, Kunming 650221, China[2]
Key Laboratory in Software Engineering of Yunnan Province, Kunming 650091, China[3]
(Email: httx133@qq.com)
(Received Aug. 24, 2018; Revised and Accepted Feb. 7, 2019; First Online Sept. 21, 2019)

Abstract

The security of cloud services is one of the most important factors to consider when users choose cloud services. An objective and quantitative measure of cloud services security directly determines whether potential users will choose cloud services or not. Aiming at this measure problem, and on the basis of STC 1.0, ISO/IEC 25010 and CIA security requirement model in the field of information security, cloud service security attribute model (CSSAM) is built. Then, a method to figure up the weights of each attributes in CSSAM is raised based on the information entropy, information gain (IG) and other concepts and formulas. At last, introduce the weights calculation method via case analysis and prove the feasibility and correctness of CSSAM model and weights calculation method.

Keywords: Cloud Service; Information Entropy; Information Gain (IG); Security Attributes

1 Introduction

Authority agency Forrester pointed out that, by 2020, the revenues global SaaS of software will reach $132.6 billion with an average increase of 9.14% each year. IDC indicated that the public cloud spending will be twice as much as the revenue reaching $127.5 billion [23]. Enterprises as the main driving force for Cloud Computing not only access to the opportunities from cloud, such as cost advantages, strategic flexibility, focus on core competencies, access to specialized resources and quality improvements, but salient risks as well, which include: performance risks, economic risks, strategic risks, security risks and managerial risks [3]. For enterprises, especially the small and medium sized enterprises (SMEs), besides the cost, whether cloud services are chosen or not depends on the questions on functionality, usability, integration, security, efficiency [35], real-time, maintainability [9] and so on.

With the rapid development of information technology and network, an increasing number of cloud services have been developed, and remarkable achievements have been made in aspects of functionality, usability and other aspects, and the problem hindering the further development of cloud services ultimately falls on the aspect of cloud service security. As the existing security evaluation and measurement methods are all provided by cloud service producers, agents or cloud service consumers, potential users or new consumers have to search services based on others' comments. Generally, the security of cloud service is positively related to its cost and the higher the security is, the higher the cost will be. But security is made up of many attributes and the highest score service does not always mean the most suitable for specific enterprises. Actually, for different enterprises, there exist differences in the demand for security, and the cloud service product scoring results from outsides are too subjective to reflect the differences in security between different enterprises. Thus, its reference value is greatly questioned.

Based on the problems above, the objective of this paper is to provide a customized method to measure the security of cloud services according to user's security requirements instead of evaluation of service providers or agents.

To achieve the objective, the following tasks will be accomplished in this paper:

1) Divide the personnel involved in cloud computing into two roles: potential users and external personnel. Potential users provide the data reflecting their requirements which will be used to calculate the attributes weights.

2) Collect and collate attributes of cloud service security and build the cloud service security attribute model, CSSAM.

3) Propose the measure method and steps based on entropy and information gain (IG).

4) Verify and validate the correctness and feasibility of the measurement method.

The structure of this paper is as follows: Section 1: The background, content and significance to cloud service security; Section 2: A survey of cloud service, cloud service security and cloud service security measurement methods; Section 3: Definitions, principles and formulas; Section 4: Cloud service security attribute index model CSSAM and specific security measure method; Section 5: A case study and analysis to the feasibility and correctness of the CSSAM model and the proposed measurement method; And Section 6: Conclusion.

2 Related Work

2.1 Cloud Computing and Cloud Computing Security

At present, cloud computing has become one of the research hotspots in the computer field [11]. In addition to the development of cloud services themselves, the cloud service security has increasingly become a hot issue in cloud computing research. The development history of information security has proved that major changes in information technology will directly affect the development process of information security [5]. Cloud Computing, with dynamic distribution of services as its main technical feature, is a major change in the field of information technology, which is bound to have a huge impact on the security sector. Main researches in this major change include trusted cloud computing [1,26,27,36], cloud service data security [10,12,17,21,22] and cloud service resource management security [4,5,18,24,32].

All the researches above focused on either functionality of cloud service or any other single security aspect, and lack overall consideration on cloud computing and cloud computing security. Cloud service security is a systematic project. If any security hole or defect is found, there is no security at all. From above, researches on overall security measurement and detection are also an important aspect to study the security of cloud services. In this regard, this paper proposes a method for measuring the overall security of cloud services which provides reliable, reliable, objective and quantitative basis for users to select products suitable for their own security needs. Meanwhile, the measurement method proposed can also provide reference for cloud service developers to improve their cloud service security.

2.2 Reviews on Cloud Service Security Measurement and Evaluation

For the measurement or evaluation of cloud service security, the existing research results are more focused on the measurement or evaluation of security risks. Chhabra and Tangja [6] discussed the risk problems of cloud computing security from the three levels of IaaS, PaaS and SaaS. Tanimoto et al. [30] started their research from the user's point of view and listed the cloud computing security risks that users are concerned about. Meanwhile, for the assessment, solution and response plans of cloud computing security, Saripalli and Walters [28] put forward a framework for cloud computing security risk assessment. Yu and Ji [34] have made a detailed analysis to the purpose, objectives, and risk assessment business processes based on roles, structures, and the context of information systems and proposed a risk assessment method for information system security oriented to business process. Other measure methods like Gini coefficient [16] represents the uncertainty of a randomly selected sample in a subset and can only assess the overall uncertainty of risk.

In order to implement security risk measurement and assessment and identify factors affecting risks, quantification of all factors is a must. The research results in this respect are as follows: risk value model VaR (Value at Risk) [37], actuarial model [7], coherent risk measurement [8, 25], information entropy and Markov chain model [2, 33] and so on. These models and methods provide important reference value for risk measurement and evaluation. In the aspect of risk assessment, existing research results [10,20,31] are mainly for single risk or similar risk analysis. As security analysis is a holistic project, single analysis is bound to lack extensive and relevant analysis, and its effects in system security are limited. In other researches, the results [14, 15, 29] pay more attention to technical risks and the analysis in uncertainty and quantitative of risk factors is inadequate. As a result, the analysis methods lack the objectivity and accuracy to evaluate the overall security of the system.

In summary, on the basis of summarizing the traditional information security demand model and previous research results, this paper starts with the security attributes that affect cloud services, and proposed cloud service security attribute mode, CSSAM, whose initial scores come from the potential users themselves. So, the model proposed reflects the different users' needs for cloud service security. Besides, scores stem from overall security performance of different products, which suggest the integrity principle of cloud service security.

3 Definitions

3.1 Roles in Cloud Computing

In this paper, roles in cloud computing are divided into two kinds: one is planning to use cloud services, named the potential cloud service user (PCSU). This kind of users is up to finish the questionnaires to compute the weights of security attributes. The other one kind is the people who is familiar with cloud services and we named this kind external personnel. The role of external personnel is made up of cloud service user (CSU), cloud service producer (CSP) and cloud service agent (CSA). Based on their own conditions or advantages, such users scored the corresponding cloud service products according to secu-

Table 1: Roles in cloud services

Categories	Role Names	Specific Descriptions
Potential users	potential cloud service user (PCSU)	Organizations or enterprises that have not directly used cloud services which hope to achieve the highest cost performance on the basis of meeting its basic functional requirements. They are always small and medium enterprises. PCSUs are always the ones who will start or add a new business on the internet. After using the services, they will become CSU which belongs to the kind of external personnel.
External personnel	Cloud service producer (CSP)	Providers of cloud services which provide differentiated services of various types and levels, and have flexible charging mode. They can be large scale enterprise and can also be single programmer. When developing a new service, they may also use others' services and then they can also become PCSUs.
	Cloud service user (CSU)	Users who used or are using or experiencing cloud services. The service evaluation data submitted by CSU is an important reference for evaluating cloud services. When new requirements come, they may also become PCSUs to find suitable services.
	Cloud service agent (CSA)	Middlemen of cloud services, who lie between CSP and PCSU. Middlemen have a clear understand of the products from CSP and the needs from PCSU. Due to the drive of value and interest, the evaluation of cloud service products from CSA is more subjective. Usually, CSA is just a service providing platform like apple store.

rity attributes. Descriptions about all roles are shown in Table 1.

3.2 Information Entropy and Information Gain

1) Information entropy

Shannon introduced physical entropy into information theory and defined the magnitude of information, which is used to measure the amount of information and named information entropy. Simply, information entropy is a tool to describe the uncertainty of information before and after communication, and its definition [13] is as follows: Definition (information entropy) Let X be a discrete random variable, and n is the number of its possible values, that means X = x1, x2, ..., xn. For each xi, its probability value is P(xi) and:

$$H(X) = -\sum_{i=1}^{n} P(x_i) \log P(x_i). \quad (1)$$

$H(X)$ is called information entropy of discrete random variable X.

2) Conditional entropy

Definition (Conditional entropy) [19] Let (X, Y) be discrete variable and its joint probability distribution is:

$$P(X = x_i, Y = y_j) = p_{ij}, \quad i = 1, 2, \cdots, n; j = 1, 2, \cdots, m. \quad (2)$$

Conditional entropy H (Y—X) indicates the uncertainty of Y of a random variable under the condition of known random variable X. Actually, conditional entropy H(Y—X) is the mathematical expectation of conditional probability distribution entropy of Y to X under given X condition and the formula is:

$$H(Y|X) = \sum_{i=1}^{n} p_i H(Y|X = x_i). \quad (3)$$

Among it, $p_i = P(X = x_i)$, $i = 1, 2, \cdots, n$.

3) Information gain

Information gain [37] is also called mutual information which indicates the reduction degree of uncertainty of Y when X is confirmed.

Definition (Information gain) For a given set D, which characteristic X is included, the information gain G(D—X) is the difference between the overall information entropy H(D) and the conditional entropy H(D—A).

$$G(D, A) = H(D) - H(D|A). \quad (4)$$

Obviously, $H(D) \geq H(D|A), G(D, A) \geq 0$.

Information gain indicates the effect of an attribute or feature $x_i \in X$, $i = 1, 2, \cdots, n$ on the overall uncertainty of the system. For each attribute of cloud service security, information gain represents the impact of a security attribute on the overall security of the cloud service. So, in this paper, information gain is used to represent as the weight of each security attribute in cloud service. As the criteria for evaluation of cloud services to be selected, the weights can be used to measure the security of cloud services and help PCSU to find the suitable services.

4 Cloud Service Security Measurement Model and Method

4.1 CSSAM

In the traditional information security field, security is embodied in the CIA security requirement model which includes: confidentiality, integrity and availability. Meanwhile, as the extension of traditional information security, other security attributes summarized from Software Trustworthiness Classification Specification and ISO/IEC 25010, such as controllability, non-repudiation, authentication, auditability, survivability and testability, should also be considered.

On the basis of STC 1.0, iso/iec 25010 and CIA security requirement model, we analyzed the characteristics of cloud computing services and users' requirements for security attributes, and then proposed cloud computing security attribute model CSSAM. Definition (CSSAM) CSSAM is a 9-tuple, CSSAM = $< f_1, f_2, f_3, f_4, f_5, f_6, f_7, f_8, f_9 >$. For each f in CSSAM, it represents the 9 security attributes separately and the specific descriptions of attributes are shown in Table 2.

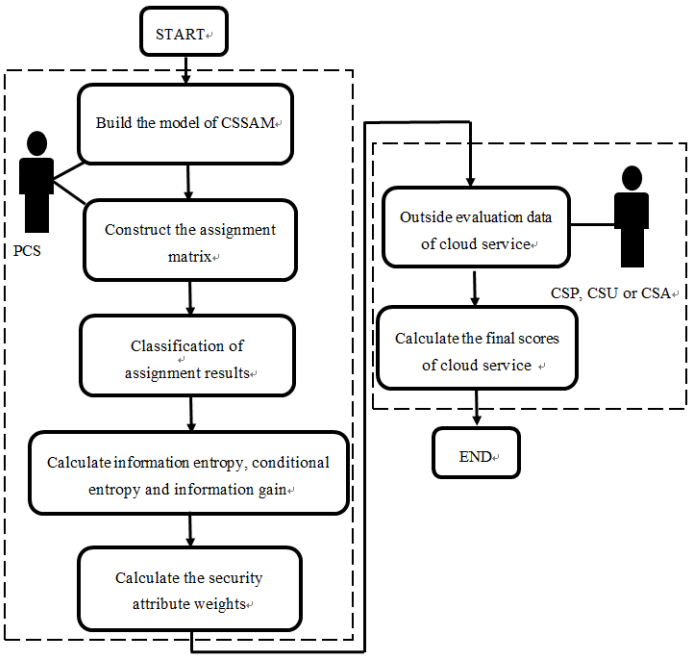


Figure 1: The proposed scheme

4.2 Measurement Method of Cloud Service Security

After building the model of CSSAM, measurement method is given. Method consists of two parts. Part one: according to the PCSU's security requirement, compute the weights on each attribute in CSSAM. The participating role in this part is the staff in PCSU, which helps finish questionnaires for scoring the attributes based on their own business. Part two: compute the final results of cloud service for the PCSU by multiplying weights getting from part one and scores of cloud services getting from external personnel outside PCSU, including CSP, CSU and CSA. Specific steps are shown in Figure 1.

Part one: Calculate the weights of attributes in CSSAM for PCSU. Steps are as follows:

Step 1: According to relevant definitions and descriptions in section 4.1, build CSSAM. Then, design the questionnaire and design the score and grading standard for each attribute in the questionnaire. Potential users PCSU design scoring rules according to their needs, which are going to be used to establish attribute weights. In addition, if having new requirements for security attributes, PCSUs can modify the specific attributes in CSSAM. An example of grading standard table is shown in Table 3. The criteria for comparison among the attributes are based on the scoring method in AHP, and fi means one of the attributes.

Step 2: Statistics the results of each questionnaire after PCSUs finished and use min-max standardization method and Equation (5) to calculate the standard results of security attributes.

$$Y_i = \frac{X_i - X_{min}}{X_{max} - X_{min}} \tag{5}$$

Among the formula, Y_i represents comprehensive score after standardization of each security attribute, and X_{max} represents the maximum value, and X_{min} the minimum value. The results constitute an assignment matrix A:

$$\begin{aligned} A_{n\times m} &= \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1m} \\ a_{21} & a_{22} & \cdots & a_{2m} \\ \cdots & \cdots & \cdots & \cdots \\ a_{n1} & a_{n2} & \cdots & a_{nm} \end{bmatrix} \\ &= (\alpha_1, \alpha_2, \cdots, \alpha_m). \end{aligned} \tag{6}$$

In the matrix, n represents the number of products used to get PCSUs' security requirements' data; m represents the number of security attributes; and a_{ij} represents the standardization result of attribute j in product i. Simply speaking, it is the target values of system security attributes based on their daily work. $\alpha_k = (a_{1k}, a_{2k}, \cdots, a_{nk})^T$ and $k = 1, 2, \cdots, m$. Add the values of each row in the matrix to get the security score of each product:

$$\begin{aligned} \beta &= (b_1, b_2, \cdots, b_n)^T \\ &= \left[\sum_{k=1}^{m} a_{1k} \sum_{k=1}^{m} a_{2k} \cdots \sum_{k=1}^{m} a_{nk}\right]^T \end{aligned} \tag{7}$$

Then, the original assignment matrix is trans-

Table 2: Security attributes in CSSAM

Nos.	Name of the attributes	Specific Descriptions
f_1	Confidentiality	Ensure that information resources are accessed only by legitimate entities (such as users, processes, etc.) and do not leak information to unauthorized entities.
f_2	Controllability	Ensure that information managers can carry out necessary control and management of information and content transmitted. Authentication, authorization, and monitoring of information and information systems to ensure the authenticity of an entity (user, process, etc.).
f_3	Integrity	Ensure that information resources can only be modified by authorized or authorized means, and not to be accidentally or deliberately altered or forged during storage or transmission.
f_4	Non-Repudiation	Ensure that the sender of information cannot deny part of the information or information that has been sent out, and the receiver of information cannot deny part of the information or information that has been received.
f_5	Survivability	Ensure that computers continue to provide core services in the face of various attacks or errors, and ensure to be able to recover all services in time, and key business functions maintained.
f_6	Auditability	Ensure the behavior of users which can be verified by using security mechanisms such as auditing, monitoring, and non-repudiation, and provide investigation evidence and means for network security problems.
f_7	Availability	Ensure that information resources can be accessed by legitimate users and can be used according to the required characteristics without being denied service.
f_8	Authentication	Ensure that information users and information providers are real claims, preventing attacks from impersonation and repetition.
f_9	Testability	Testability is the ability of software to detect faults and isolate and locate faults and ability of design and testing execution under certain time and cost.

Table 3: An example of marking standard

Levels	Values	Description
Extremely important	8-10	In CSSAM, fi is extremely important
Specially important	6-8	In CSSAM, fi is very important
Very important	4-6	In CSSAM, fi is obviously important
Fairly important	2-4	In CSSAM, fi is a little more important
Unimportant	0-2	In CSSAM, two attributes have the same or similar importance.

formed into $A'_{n\times m}$:

$$A'_{n\times m} = (\alpha_1, \alpha_2, \cdots, \alpha_m, \beta). \tag{8}$$

Step 3: According to the value of β in matrix $A'_{n\times m}$, every attribute has got a security level, which is used to calculate the information entropy in the next step. The specific levels can be set based on the actual needs of PCSU, and security attribute levels can refer to the table in Step 1, and the levels of β values are shown in Table 4.

Table 4: An example of grading

Levels	Scores
UNSAFE	0-30
MEDIUM	30-60
SAFE	60-90

Step 4: The matrix values getting from Step 2 and Step 3, are used to calculate information entropy H(β) using Formula (1) and Formula (3) is used to calculate conditional entropy H($\beta|\alpha_i$) of each security attribute, and Formula (4) is used to calculate information gain G(β, α_i) of each security attribute.

Step 5: Normalize the values of security attributes information gain $G(\beta, \alpha_i)$ getting from Formula (4) and security attribute weight γ_i is obtained. γ_i comes from PCSU's staff, so it reflects the security requirements of PCSU.

Part Two: According to the weights getting from part one, measure the service to be chosen for PCSU. Steps are as follows:

Step 6: Seek evaluation data of cloud services to be selected for PCSU. Usually, the data can be from the roles of external personnel, and the initial data should be standardized by min-max standardization method, and security attribute scores $\delta = (e_1, e_2, \cdots, e_m)$ of some cloud service from external personnel are achieved.

Step 7: For PCSU, the formula for calculating the security score S of the cloud service to be measured is:

$$S = \delta \times \gamma^T.$$

In the formula, γ^T is the transpose of γ and γ is the collection of the 9 security attributes weights and $\gamma = (\gamma_1, \gamma_2, \gamma_3, \gamma_4, \gamma_5, \gamma_6, \gamma_7, \gamma_8, \gamma_9)$.

5 Case Analysis

5.1 Case

To expand its business, some company named E intends to build an online sales management system. As restricted by capital and technology, after investigation, E decides to build their new system by hiring cloud services.

The company used to do online selling in office supplies (B2C), but it found a business opportunity in marketing country agricultural commodities for urban communities (B2B and B2C). This company had to reorganize all the business, transferring to mobile phone client. Owing to the scale, cost and technique, it decided to apply cloud services in order to implement the new business. As the main trade product, fresh product needs more strict standards in technique. If information transmission and processing fail, it will pose a great economic threat to the company. Thus, it needs more strict demands in availability and survivability. On the other hand, the main business is to implement a vital link between village head and property company managers. As for farm product providers or communities, they need to do business through village head or property company managers. In order to guarantee each participant's rights, the company has much higher demands in controllability and auditability of information.

The method proposed in this paper is used to measure the four services for the E company to choose the service which will most satisfy their own security requirements. Measurement processes are as follows:

Step 1: Establish the model of CSSAM and PCSU can select their own security attributes and design the questionnaires for their staff. The grading standard can follow Table 3.

Step 2: Formula (5) is used to calculate the results from PCSU staff questionnaires and put the results into the scoring matrix. Using Formula (7), β is obtained by adding the scores of the 9 attributes and filled in the matrix as well. The scoring matrix A is as shown in Table 5. In Table 5, rows represent the products chosen for test and the columns represent the 9 attributes. The data in the table were scored by 10 front-line staff of E company according to AHP rules for 9 attributes of 15 software products. The company's preference for security attributes can be reflected by the results of scoring different software security by front-line personnel within the company.

Table 5: Scoring matrix

	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8	f_9	β
1	10	9	8	6	7	5	6	8	8	67
2	6	8	7	4	3	6	4	5	3	46
3	7	5	6	4	5	3	2	5	5	42
4	6	3	6	3	1	0	1	4	3	27
5	6	6	8	7	7	4	3	6	8	55
6	8	7	9	9	8	6	8	6	7	68
7	9	9	8	9	8	6	8	5	8	69
8	10	8	9	8	9	8	4	6	6	68
9	5	5	8	6	6	8	4	3	5	50
10	8	5	7	9	7	3	2	5	4	50
11	8	8	10	9	9	4	6	7	9	71
12	7	9	6	5	5	2	5	3	7	50
13	6	1	5	3	3	0	1	3	5	27
14	8	4	7	8	7	4	4	4	8	54
15	6	4	7	6	8	2	4	5	8	50

Step 3: Scores in Table 5 are divided into different grades according to the standards in Table 3 and Table 4 and the dividing results are as shown in Table 6. In the table, EI means extremely important; SI means specially important; VI means very important; FI means fairly important; UI means unimportant.

Step 4: Formula (1) is used to calculate information entropy $H(\beta) = 0.97$. Formula (3) is used to calculate the conditional entropy $H(\beta|\alpha_i)$ of each security attribute in Table 6 and then, information gain $G(\beta, \alpha_i)$ of each attribute is got by using Formula (4). The results are as shown in Table 7.

Step 5: Normalize the results of $G(\beta, \alpha_i)$ in Table 7 and weight of each security attribute is obtained. The results are shown in Table 8. γ_i means the weight of attribute f_i.

Table 6: Grades of each attributes

	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8	f_9	β
1	EI	EI	EI	VI	SI	VI	SI	EI	EI	SAFE
2	SI	EI	SI	VI	FI	SI	VI	VI	FI	MEDIUM
3	SI	VI	SI	VI	VI	FI	FI	VI	VI	MEDIUM
4	SI	FI	SI	FI	UI	UI	UI	VI	FI	UNSAFE
5	VI	VI	EI	SI	SI	VI	FI	SI	EI	MEDIUM
6	EI	SI	EI	EI	EI	SI	EI	SI	SI	SAFE
7	EI	EI	EI	EI	SI	SI	EI	VI	SI	SAFE
8	EI	EI	EI	SI	EI	EI	VI	SI	SI	SAFE
9	VI	VI	EI	SI	SI	EI	VI	FI	VI	MEDIUM
10	SI	VI	SI	EI	SI	FI	FI	VI	VI	MEDIUM
11	EI	EI	EI	EI	EI	VI	SI	SI	EI	SAFE
12	SI	EI	SI	VI	VI	FI	VI	FI	SI	MEDIUM
13	SI	UI	VI	FI	FI	UI	UI	FI	VI	UNSAFE
14	EI	FI	SI	EI	SI	VI	VI	VI	SI	MEDIUM
15	SI	VI	SI	SI	SI	FI	VI	VI	EI	MEDIUM

Table 7: The results of conditional entropy and information gain

Security Attributes	$H(\beta\|\alpha_i)$	$G(\beta,\alpha_i)$
f_1	0.46	0.51
f_2	0.35	0.62
f_3	0.47	0.50
f_4	0.52	0.45
f_5	0.28	0.69
f_6	0.40	0.57
f_7	0.18	0.79
f_8	0.65	0.32
f_9	0.65	0.32

Table 8: Weights of each attribute indexes

Security Attributes	γ_i
f_1	0.11
f_2	0.13
f_3	0.10
f_4	0.09
f_5	0.14
f_6	0.12
f_7	0.17
f_8	0.07
f_9	0.07

Step 6: According to the attributes in CSSAM, seek the security evaluation data of the four cloud services (CS1, CS2, CS3, CS4) and Table 9 is established. The data in the table were the mean value which were calculated from scores of external personnel (include 10 CSAs, 10 CSPs and 30 CSUs) according to AHP rules for nine attributes of the four products. Each score means the sum of each attribute of each cloud service product.

The scores of the four cloud services are based on the AHP rules of 50 external personnel (include 10 CSAs, 10 CSPs and 30 CSUs). Because of the exclusiveness of security attributes, the strength of some attributes will inevitably lead to the decline of other attributes. Four service products have different emphasis on nine security attributes. CS1 focuses on authentication, non-repudiation, survivability and testability; CS2 focuses on availability and survivability; CS3 focuses on confidentiality, controllability availability and authentication; CS4 focuses on availability, testability, survivability, auditability and controllability. As CS2 only performs well in availability and survivability, other attribute scores are very low, resulting in the lowest total score in the four products.

Table 9: Scores getting from external personnel

	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8	f_9	Scores
CS1	8	6	8	9	9	7	5	10	9	70
CS2	6	5	3	3	9	5	10	6	5	52
CS3	10	10	7	5	5	5	10	9	8	69
CS4	4	9	3	6	9	9	10	8	10	67

Step 7: The four services' final scores S for E company

will be gotten using Formula (9) and results are shown in Table 10.

From Table 8, we can get that, f_7, f_5 and f_2 are the most concerned security factors, that means, compared with others, E company pay more attention to the availability, survivability and controllability. And attributes like authentication and testability do not have much concern.

5.2 Comparison and Analysis

In the case, we have calculated the weights of the 9 attributes of the E company. According to the scores of four cloud services given by external personnel staff, the results that meet the company's security requirements are calculated, as is shown in Table 10. We choose two commonly used weight calculation methods: average method and AHP scoring method, and compare them with information gain method proposed in this paper.

1) Average method (AVG). Assuming that each security attribute has the same weight, that means $\gamma_i = 1/9$, $i = 1, 2, \cdots, 9$, and $\gamma_{avg}^T = (0.11, 0.11, \cdots, 0.11)$.

2) AHP scoring method (AHP). Collate the scoring results of 20 external personnel (include 5 CSAs, 5 CSPs and 10 CSUs) and obtain the weights of the 9 security attributes, $\gamma_{AHP}^T = (0.20, 0.12, 0.18, 0.02, 0.16, 0.03, 0.20, 0.04, 0.05)$.

According to Formula (9), the scores of four CSs using different weight calculation methods are calculated, and the results are shown in Table 11 and Table 12. In the two tables, S represents the total score of four cloud service products calculated by the two methods.

Combining the results calculated in Table 10, the results of scoring four cloud service products by three methods are shown in Table 13 and Figure 2.

In Figure 2(a), (b) and (c) are the final scores of the four cloud services calculated by multiplying the weights obtained by using information gain, average value and AHP method with the results of external personnel scoring in Table 9. Figure 2(b) directly reflects the external measurement results of the security of four products. External recognition of the security of four products is in the figure, that is, $CS1 > CS3 > CS4 > CS2$. This only shows the degree of recognition of various cloud service products by the outside world, but does not reflect the security needs preferences of specific users.

Figure 2(c) is the weight of each security attribute derived by the external staff associated with cloud services according to their awareness of security attributes. According to the objective score data of each cloud service, the final result is $CS3 > CS1 > CS4 > CS2$. This reflects the industry's recognition of security attributes. Combining the scores of four cloud service products, the professional evaluation of four cloud services by professionals is given. Its reference value is higher than the average method. In this method, CS3 products perform best in attributes that experts value, so it gets obvious high scores. The service of CS3 may have got a good score in most situations, but still may not be perfect for a special application scenario.

Figure 2(a) is the result of the method presented in this paper. The weights are calculated by the internal personnel analysis, which reflects the user's own security needs. Combined with the evaluation of four cloud services given by the outside world, the results not only reflect the objective degree of product security, but also reflect the degree of conformity of the cloud services to the company's security needs.

For company E, the result calculated with the method proposed in this paper is: $CS4 > CS1 > CS3 > CS2$. Obviously, CS4 has the highest score and should be the most suitable service for E. Compared with the scores from other methods, the calculated scores are more consistent with E's own business security requirements, and the advantages of CS4 focuses on the attributes of controllability, survivability, auditability, availability and testability, which meet the needs of E's security requirements in the practical application process.

6 Conclusion

As one of the most crucial attributes, the security of cloud services also becomes the most dominant element for traditional users to choose cloud services. An objective and quantitative measure of cloud services security directly determines whether potential users will choose cloud services. In the existing literature, researchers only measured and evaluated the product itself and ignored the specific security needs of users. Therefore, for specific users, existing methods will not meet their customized security requirements. Regarding the objective quantitative measure problem, different users in different field or one user in different application environments have diverse demand in cloud services security.

To solve the problem above, the following contributions have been made: (1) We have introduced the roles and roles' classification in cloud computing, containing potential users (PCSU), cloud services providers (CSP), cloud services users (CSU) and cloud services agents (CSA). (2) On the basis of STC 1.0, ISO/IEC 25010 and CIA security requirement model in the field of information security, a cloud service security attribute model CSSAM has been proposed, which includes 9 security attributes for measurement. (3) A method based on the entropy, information gain (IG) and other concepts and formulas has been raised to define weights of each attributes in user CSSAM. (4) A case study has been introduced which proved the feasibility and correctness of CSSAM model and weights calculation method in practice. Our work has enriched the application scenarios of information entropy and information gain (IG) theory. In practical application, it has also made some contributions to objectively measure cloud services and help SMEs choose suitable cloud ser-

Table 10: Final scores of 4 cloud services

	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8	f_9	S
CS1	0.83	0.79	0.80	0.84	1.30	0.84	0.83	0.67	0.60	7.50
CS2	0.64	0.63	0.31	0.28	1.30	0.59	1.66	0.40	0.34	6.16
CS3	2.04	1.31	0.69	0.45	0.72	0.57	0.33	0.61	0.56	7.27
CS4	0.43	1.18	0.31	0.52	1.30	1.05	1.66	0.54	0.65	7.64

Table 11: AVG scores of 4 cloud services

	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8	f_9	S
CS1	0.89	0.67	0.89	1.00	1.00	0.78	0.56	1.11	1.00	7.89
CS2	0.67	0.56	0.33	0.33	1.00	0.56	1.11	0.67	0.56	5.78
CS3	1.11	1.11	0.78	0.56	0.56	0.56	1.11	1.00	0.89	7.67
CS4	0.44	1.00	0.33	0.67	1.00	1.00	1.11	0.89	1.11	7.56

Table 12: AHP scores of 4 cloud services

	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8	f_9	S
CS1	1.60	0.70	1.44	0.17	1.44	0.22	1.00	0.42	0.47	7.45
CS2	1.20	0.59	0.54	0.06	1.44	0.15	1.99	0.25	0.26	6.48
CS3	2.00	1.17	1.26	0.09	0.80	0.15	1.99	0.38	0.42	8.27
CS4	0.80	1.06	0.54	0.11	1.44	0.28	1.99	0.33	0.52	7.07

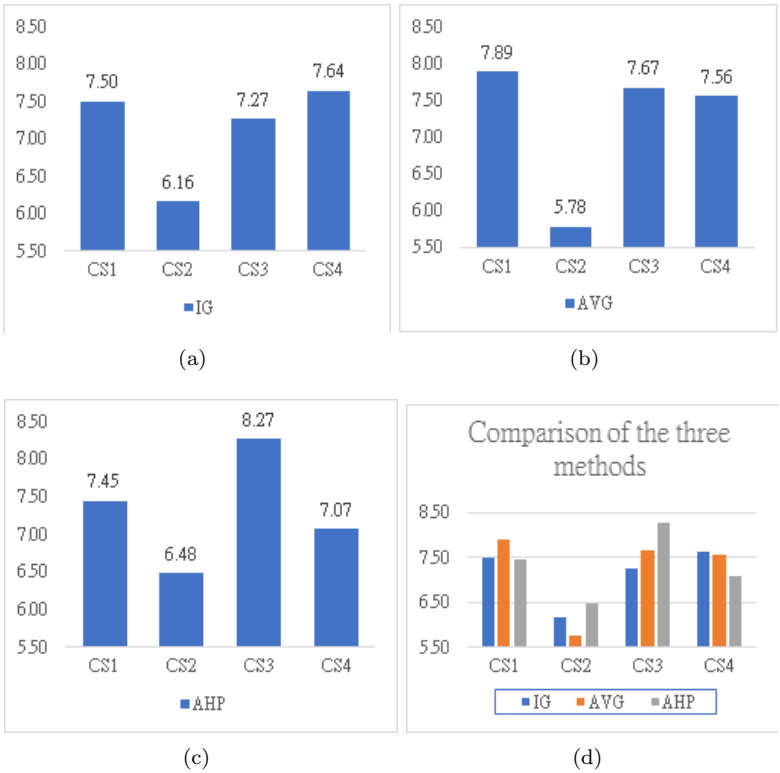


Figure 2: The comparison of the 3 methods

Table 13: AVG scores of 4 cloud services

	CS1	CS2	CS3	CS4
IG	7.50	6.16	7.27	7.64
AVG	7.89	5.78	7.67	7.56
AHP	7.45	6.48	8.27	7.07

vices. While the attributes in CSSAM will develop or change with the environment and society, in the future, cloud computing and cloud services will continue to be further studied, and we will further improve and refine security attributes in CSSAM and give more reasonable and detailed security attributes and descriptions to better measure the security of cloud services and provide more reliable basis for PCSU to selected suitable cloud services.

Acknowledgments

This work was supported by National Natural Science Foundation of China (Nos.61379032, 61763048, 61263022, 61303234, 61662085), National Social Science Foundation of China (No.12XTQ012), Science and Technology Foundation of Yunnan Province (No.2017FB095), Yunnan Province Applied Basic Research Project(No.2016FD060), Science Research Project of Yunnan Education (Nos.2017ZZX001, 2017ZZX227), Key Project of Scientific Research of Yunnan Education (2015Z018), Provincial Scientific and Technological Innovation Team Project of Yunnan University (2017HC012), the 18th Yunnan Young and Middle-aged Academic and Technical Leaders Reserve Personnel Training Program (No.2015HB038). The authors would like to thank the anonymous reviewers and the editors for their suggestions.

References

[1] D. S. AbdElminaam, "Improving the security of cloud computing by building new hybrid cryptography algorithms," *International Journal of Electronics and Information Engineering*, vol. 8, no. 1, pp. 40-48, 2018.

[2] M. H. R. Al-Shaikhly and H. M. El-Bakry and A. A. Saleh, "Cloud security using markov chain and genetic algorithm," *International Journal of Electronics and Information Engineering*, vol. 8, no. 2, pp. 96-106, 2018.

[3] A. Benlian, and T. Hess, "Opportunities and risks of software-as-a-service: Findings from a survey of it executives," *Decision Support Systems*, vol. 52, no. 1, pp. 232-246, 2012.

[4] P. Bonatti, S. D. C. di Vimercati, P. Samarati, "An Algebra for Composing Access Control Policies," *ACM Transactions on Information and System Security*, vol. 5, no. 1, pp. 1-35, 2002.

[5] S. Bulusu, and K. Sudia, "A Study on Cloud Computing Security Challenges," *DiVA Portal*, 2012. (`https://www.diva-portal.org/smash/get/diva2:830115/FULLTEXT01.pdf`)

[6] S. P. Chandran, and M. Angepat, "Cloud computing: Analysing the risks involved in cloud computing environments," *Ornitologia Neotropical*, vol. 12, 2001.

[7] M. Eling, and N. Loperfido, "Data breaches: Goodness of fit, pricing, and risk measurement," *Insurance Mathematics & Economics*, vol. 75, pp. 126-136, 2017.

[8] H. Föllmer, and I. Penner, "Consistent risk measures and a non-linear extension of backwards martingale convergence," in *Interdisciplinary Mathematical Sciences & Festschrift Masatoshi Fukushima* pp. 183-202, 2015.

[9] J. Espadas, A. Molina, G. Jiménez, M. Molina, R. Ramírez, D.Concha, "A tenant-based resource allocation model for scaling; Software-as-a-service applications over cloud computing infrastructures," *Future Generation Computer Systems*, vol. 29, no. 1, pp. 273-286, 2013.

[10] I. A. T. Hashem, I. Yaqoob, N. B. Anuar, S. Mokhtar, A. Gani, and S. U. Khan, "The rise of "big data" on cloud computing: Review and open research issues," *Information Systems*, vol. 47, no. C, pp. 98-115, 2015.

[11] C. Hoffa, G. Mehta, T. Freeman, E. Deelman, K. Keahey, B. Berriman, and J. Good, "On the use of cloud computing for scientific workflows," in *IEEE Fourth International Conference on Escience*, 2008. DOI: 10.1109/eScience.2008.167.

[12] L. C. Huang, L. Y. Tseng, M. S. Hwang, "A reversible data hiding method by histogram shifting in high quality medical images," *Journal of Systems and Software*, vol. 86, no. 3, pp. 716-727, 2013.

[13] E. T. Jaynes, "Information theory and statistical mechanics," *Physical Review*, vol. 106, no. 4, pp. 620-630, 1957.

[14] C. Joshi, and U. K. Singh, "Information security risk management framework for university computing environment," *International Journal of Network Security*, vol. 19, no. 5, 2017.

[15] M. Jouini, and L. B. A. Rabai, "Comparative study of information security risk assessment models for cloud computing systems," *Procedia Computer Science*, vol. 83, pp. 1084-1089, 2016.

[16] W. Kalmijn, Gini Coefficient: Springer Netherlands, 2014.

[17] S. Kalra, and S. K. Sood, "Secure authentication scheme for IoT and cloud servers," *Pervasive & Mobile Computing*, vol. 24, pp. 210-223, 2015.

[18] C. C. Lee, C. H. Liu, and M. S. Hwang, "Guessing attacks on strong-password authentication protocol," *International Journal of Network Security*, vol. 15, no. 1, pp. 64-67, 2013.

[19] H. Li, *Statistical Learning Method: Tsing University Press*, 2012. (`https://www.amazon.com/Statistical-learning-methods-LI-HANG/dp/7302275955`)

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

[21] L. Liu, W. Kong, Z. Cao and J. Wang, "Analysis of one certificateless encryption for secure data sharing in public clouds," *International Journal of Electronics and Information Engineering*, vol. 6, no. 2, pp. 110-115, 2017.

[22] C. Liu, C. Yang, X. Zhang, and J. Chen, "External integrity verification for outsourced big data in cloud and IoT," *Future Generation Computer Systems*, vol. 49, no. C, pp. 58-67, 2015.

[23] Z. Ma, R. Jiang, M. Yang, T. Li, and Q. Zhang, "Research on the measurement and evaluation of trusted cloud service," *Soft Computing*, vol. 22, no. 6, pp. 1-16, 2016.

[24] J. Mclean, "The specification and modeling of computer security," *Computer*, vol. 23, no. 1, pp. 9-16, 1990.

[25] S. Mitra, Sovan, "Efficient option risk measurement with reduced model risk," *Insurance Mathematics & Economics*, vol. 72, pp. 163-174, 2017.

[26] A. R. Sadeghi, T. Schneider, and M. Winandy, "Token-based cloud computing: Secure outsourcing of data and arbitrary computations with lower latency," in *International Conference on Trust and Trustworthy Computing*, 2010. DOI:10.1007/978-3-642-13869-0.

[27] N. Santos, K. P. Gummadi, and R. Rodrigues, "Towards trusted cloud computing," in *International Conference on Hot Topics in Cloud Computing*, 2009. (`https://people.mpi-sws.org/~gummadi/papers/trusted_cloud.pdf`)

[28] P. Saripalli, and B. Walters, "Quirc: A quantitative impact and risk assessment framework for cloud security," in *IEEE International Conference on Cloud Computing*, 2010. DOI: 10.1109/CLOUD.2010.22.

[29] F. U. Sha, Y. Z. Xiao, and M. H. Liao, "An approach for campus information systems security risk assessment based on fuzzy set and entropy weight," *Information Science*, 2013. (`http://en.cnki.com.cn/Article_en/CJFDTotal-QBKX201309023.htm`)

[30] S. Tanimoto, M. Hiramoto, M. Iwashita, H. Sato, and A. Kanai, "Risk Management on the Security Problem in Cloud Computing," in *First ACIS/JNU International Conference on Computers, Networks, Systems and Industrial Engineering*, 2011. DOI: 10.1109/CNSI.2011.82.

[31] J. Wang, J. Liu, and H. Zhang, "Access control based resource allocation in cloud computing environment," *International Journal of Network Security*, vol. 19, no. 2, pp. 236-243, 2017.

[32] D. Wijesekera, and S. Jajodia, "A propositional policy algebra for access control," *ACM Transactions on Information & System Security*, vol. 6, no. 2, pp. 286-325, 2003.

[33] M. Yang, R. Jiang, T. Gao, W. Xie and J. Wang, "Research on cloud computing security risk assessment based on information entropy and markov chain," *International Journal of Network Security*, vol. 20, no. 4, pp. 664-673, 2018.

[34] Z. Yu, and Z. Ji, "A survey on the evolution of risk evaluation for information systems security," *Energy Procedia*, vol. 17, 1288-1294, 2012.

[35] K. K. F. Yuen, "Software-as-a-service evaluation in cloud paradigm: Primitive cognitive network process approach," in *IEEE International Conference on Signal Processing, Communication and Computing*, 2012. DOI: 10.1109/ICSPCC.2012.6335719.

[36] X. Zhang, T. Li, X. Wang, Q. Yu, Y. Yu, and R. Zhu, "Formal Analysis to Non-Functional Requirements of Trustworthy Software," *Journal of Software*, vol. 26, no. 10, pp. 2545-2566, 2015.

[37] Z. Zhang, L. Yang, H. Li, and F. Xiang, "A quantitative and qualitative analysis-based security risk assessment for multimedia social networks," *International Journal of Network Security*, vol. 18, no. 1, pp. 43-51, 2016.

Biography

Tilei Gao is a lecturer at the school of information,Yunnan University of Finance and Economics. He is also a Ph.D candidate in system analysis and integration at the school of software at Yunnan University. His main research interests include software engineering and information management.

Tong Li is a professor and Ph. D. supervisor at school of software, Yunnan University, China. He received his Ph.D. from De Montfort University in 2007. His main research interests include software process, software engineering, etc.

Rong Jiang is a professor and Ph. D. supervisor at the school of information, Yunnan University of Finance and Economics, China. He received his Ph.D. from the school of software at Yunnan University. His main research interests include cloud computing, big data, software engineering, information management, etc.

Ming Yang, Corresponding author, is a lecturer at the school of information, Yunnan University of Finance and Economics, China. He received his Ph.D. from the school of software at Yunnan University. His main research interests include information management and data mining.

Rui Zhu is a lecturer at software school, Yunnan University, China. He received his Ph.D. from the school of software at Yunnan University. His main research interests include software process, software engineering, etc.

Security Analysis of a Three-factor Anonymous Authentication Scheme for Wireless Sensor Networks in Internet of Things Environments

Wei-Liang Tai[1], Ya-Fen Chang[2], and Po-Lin Hou[2]
(Corresponding author: Ya-Fen Chang)

Department of Information Communications, Chinese Culture University[1]
Department of Computer Science and Information Engineering,
National Taichung University of Science and Technology[2]
No. 129, Section 3, Sanmin Road, Taichung, Taiwan
(Email: cyf@nutc.edu.tw)

(Received Sept. 26, 2018; Revised and Accepted May 17, 2019; First Online June 25, 2019)

Abstract

The Internet of Things (IoT) can be applied to applications in various fields such as industry, medical care, and public security because IoT enables remote sensing and control in heterogeneous environments. Wireless sensor networks (WSNs) are an important infrastructure in IoT, where a sensor node provides the collected data to authorized users. Because of the resource-constrained nature of sensor nodes such as transmission and computational capabilities and the limited energy, how to ensure both security and efficiency of WSNs in IoT environments becomes a challenge. Recently, Li *et al.* proposed a three-factor anonymous authentication scheme by adopting a fuzzy commitment scheme and an error correction code to handle the user′s biometric data for WSNs in IoT environments. They claimed their scheme could ensure computational efficiency and achieve more security and functional features. After analyzing their authentication scheme, we find that it cannot ensure security. First, a malicious user can retrieve a sensor node′s secret and impersonate the sensor node. Second, a malicious user can acquire the sensory data without the gateway node even with a forged identity. Third, the malicious user can retrieve another legal user′s essential information for authentication and impersonate this innocent user. In this paper, how these security flaws damage Li *et al.*′s authentication scheme and further discussions will be shown in detail.

Keywords: Authentication; Elliptic Curve Cryptography; Internet of Things; Wireless Sensor Network

1 Introduction

The rise of the Internet of Things (IoT) [3] brings significant changes to people′s daily life. Because IoT enables remote sensing and control in heterogeneous environments, IoT is widely applied to applications in various fields such as industry, transportation, agriculture, medical care, military and public security such that Industry 4.0, Smart Transportation, Smart Home, Smart Medical, and Smart City can be realized. This makes people′s life more and more convenient.

In IoT applications, wireless sensors play an important role because they sense the surroundings, generate sensory data, and transmit data through heterogeneous network environments. Thus, wireless sensor networks (WSNs) are an important infrastructure in IoT, and a sensor node in WSNs provides the collected data to authorized users.

However, wireless sensors regarded as one of the most important devices in IoT are usually unattended. Researches indicate that the energy consumption of sensor nodes is proportional to the transmission distance so WSNs should be extended [1,9]. To increase the life cycle of WSNs, a gateway node and heterogeneous WSNs are introduced. In heterogeneous WSNs, sensors may possess different capacities such as transmission and computational capacities. It denotes that some sensor nodes such as the gateway nodes can transmit data over long distances, and the desired sensory data can be delivered to a backed server for further and real-time analysis. This property makes users obtain specific information quickly and make decisions as soon as possible.

Due to the resource-constrained nature of wireless sensors, such as transmission and computational capabilities and the limited energy, and the characteristics of public transmission medium, how to ensure both security and efficiency of WSNs in IoT environments becomes a tough and urgent issue. In 2013, Xue *et al.* proposed a time-based voucher-based mutual authentication and key agreement scheme for wireless sensor networks [12].

In Xue *et al.*'s scheme, the gateway node generates time credentials for each user and sensor node. With time credentials, a user, the gateway node and a sensor node can authenticate each other. Xue *et al.*'s scheme uses only simple computational operations, such as hash function and XOR (exclusive-or) operation, to comply with the resource-constrained nature of wireless sensors. However, in 2015, He *et al.* [5] showed that Xue *et al.*'s scheme is vulnerable to several attacks, offline password guessing attack, impersonation attack and modification attack. He *et al.* also proposed an improved temporal-credential-based mutual authentication and key agreement scheme with pseudo identity. In 2016, Jiang *et al.* [7] showed that Xue *et al.*'s scheme suffers from stolen smart card attack, user impersonation attack, and tracking attack. By using ECC, Jiang *et al.* also proposed an improvement based on He *et al.*'s scheme. In Jiang *et al.*'s scheme, sensor nodes only need to execute simple computational operations while a user and the gateway node need to execute ECC operations. As a result, the difficulty of elliptic curve discrete logarithm (ECDL) can increase the security level of their scheme. Meanwhile, Amin *et al.* proposed an anonymity preserving three-factor authenticated key exchange protocol for wireless sensor networks [2]. Unfortunately, Chang *et al.* showed that Amin *et al.* protocol cannot ensure user anonymity and suffers from desynchronization attack [4]. Although there are also some two-factor authentication schemes for wireless sensor networks, it has been demonstrated that the security of these two-factor authentication schemes is doubted [6, 10, 11].

Recently, Li *et al.* [8] showed that Xue *et al.*'s, He *et al.*'s, and Jiang *et al.*'s scheme commonly have the following flaws.

1) These schemes cannot detect wrong password and lack mechanisms to update password.
2) Messages are directly exchanged between a user and a sensor such that these schemes are not suitable for IoT environments.
3) These schemes are all vulnerable to known session-specific temporary attack and clock synchronization attack.

To overcome the drawbacks and preserve the advantages, Li *et al.* proposed a three-factor anonymous authentication scheme by adopting a fuzzy commitment scheme and an error correction code to handle the user's biometric data for WSNs in IoT environments with ECC and simple computational operations such as hash function and XOR operation. They claimed their scheme could ensure computational efficiency and achieve more security and functional features.

After analyzing the scheme proposed by Li *et al.*, we find that their scheme cannot ensure security as claimed. First, a malicious user can retrieve a sensor node's secret and impersonate the sensor node to deliver forged sensory data. Second, a malicious user can acquire the sensory data without the gateway node even with a forged identity. Third, the malicious user can retrieve another legal user's essential information for authentication and impersonate this innocent user. If different access rights are granted to different users, this flaw makes a privileged account compromised. The rest of this paper is organized as follows. Section 2 reviews Li *et al.*'s scheme. Security analysis and advanced discussions are given in Section 3. At last, some conclusions are drawn.

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

This section reviews Li *et al.*'s three-factor anonymous authentication scheme for WSNs in IoT environments. The notations used in Li *et al.*'s scheme are listed in Table 1. In Li *et al.*'s scheme, ECC is employed. First, the gateway node, GWN, selects an addition group G over a finite field F_p on the elliptic curve E of prime order n, where the point P is the generator. Then GWN randomly selects a number $x \in Z_n^*$ as its private key, chooses a master key K_{GWN}, and computes the public key $X = xP$. GWN keeps x and K_{GWN} secretly and makes $\{E(F_p), G, P, X\}$ public. Li *et al.*'s scheme is composed of four phases: sensor registration phase, user registration phase, login and authentication phase, and password change phase. Because password change phase is not related to our security analysis of Li *et al.*'s scheme, password change phase is omitted. The details are as follows.

Table 1: Notations used in Li *et al.*'s three-factor anonymous authentication scheme

Notation	Definition
U_i, GWN, S_j	i^{th} user, gateway node, j^{th} sensor node
ID_i/SID_j	Identity of U_i/S_j
PW_i	Password of U_i
b_i	Biometric of U_i
SC	U_i's smart card
K_{GWN}	GWN's master key
K_{GWN-S_j}	Secret key shared between GWN and S_j
$SK_i/SK_j/SK_{GWN}$	Session key computed by $U_i/S_j/GWN$
$h(.)$	A secure hash function
$C \subseteq \{0,1\}^n$	A set of codewords
$F(.)$	A fuzzy commitment scheme
$f(.)$	A decoding function
r_i, r_g, r_j	Random numbers generated by U_i,GWN and S_j, respectively
$\|$	Concatenation operation
$\oplus$	XOR operation

2.1 Sensor Registration Phase

Before S_j is deployed, GWN selects an identity SID_j and computes the secret key $K_{GWN-S_j} = h(SID_j \parallel K_{GWN})$ for S_j. Then GWN stores $\{SID_j, K_{GWN-S_j}\}$ in S_j's memory. At last, GWN deploys these sensors in a particular area to form a wireless sensor network.

2.2 User Registration Phase

When a user wants to acquire sensory data from sensor nodes, he/she has to register at GWN in the first place. The details are as follows:

Step 1. An identity ID_i and a password PW_i are selected by U_i.

Step 2. U_i generates a nonce a_i and computes $RPW_i = h(PW_i \parallel a_i)$.

Step 3. U_i imprints the biometric on a special device and gets the biometric information b_i.

Step 4. U_i submits the registration request $\{ID_i, RPW_i, b_i\}$ to GWN via a secure manner.

Step 5. Upon receiving the registration request, GWN chooses a random codeword $c_i \in C$ for U_i.

Step 6. GWN computes $F(c_i, b_i) = (\alpha, \delta) = (h(c_i), c_i \oplus b_i)$, $A_i = h(ID_i \parallel RPW_i \parallel c_i)$ and $B_i = h(ID_i \parallel K_{GWN}) \oplus h(RPW_i \parallel c_i)$.

Step 7. GWN stores $\{\alpha, \delta, A_i, B_i, X, f(.)\}$ into a smart card, SC, and issues it to U_i via a secure channel.

Step 8. GWN stores ID_i in its database and deletes other information.

Step 9. After getting SC, U_i stores a_i into it. Then, SC contains $\{\alpha, \delta, A_i, B_i, X, f(.), a_i\}$.

2.3 Login and Authentication Phase

When U_i wants to access the data collected by the sensor S_j, U_i should be first authenticated by GWN. The details are as follows:

Step 1. U_i inserts SC into a card reader and imprints the biometric b_i' on a special device.

Step 2. SC computes $c_i' = f(\delta \oplus b_i') = f(c_i \oplus (b_i \oplus b_i'))$ and checks if $h(c_i') = \alpha$. If it does not hold, this session is terminated by SC; otherwise, the imprinted biometric b_i' is verified successfully, and U_i inputs ID_i and PW_i.

Step 3. SC computes $A_i' = h(ID_i \parallel h(PW_i \parallel a_i) \parallel c_i')$ and checks if $A_i' = A_i$. If it does not hold, this session is rejected by SC; otherwise, U_i's identity ID_i and password PW_i are verified successfully by SC.

Step 4. SC chooses random numbers r_i and $s \in Z_n^*$.

Step 5. SC computes $M_1 = B_i \oplus h(h(PW_i \parallel a_i) \parallel c_i')$, $M2 = sP$, $M3 = sX = sxP$, $M4 = ID_i \oplus M_3$, $M5 = M_1 \oplus r_i$, $M6 = h(ID_i \parallel r_i) \oplus SID_j$ and $M7 = h(M_1 \parallel SID_j \parallel M_3 \parallel r_i)$.

Step 6. U_i sends the login request $\{M_2, M_4, M_5, M_6, M_7\}$ to GWN.

Step 7. After receiving the login request, GWN computes $M_3' = xM_2 = xsP$ and $ID_i' = M_4 \oplus M_3'$ and checks if ID_i' exists in the database. If it does not exist, this login request is rejected by GWN; otherwise, this phase proceeds.

Step 8. GWN computes $M_1' = h(ID_i' \parallel K_{GWN})$, $r_i' = M_5 \oplus M_1'$, $SID_j' = M_6 \oplus h(ID_i' \parallel r_i')$ and $M_7' = h(M_1' \parallel SID_j' \parallel M_3' \parallel r_i')$ and checks if $M_7' = M_7$. If it does not hold, this session is terminated by GWN; otherwise, GWN generates a random number r_g.

Step 9. GWN computes $K'_{GWN-S_j} = h(SID_j' \parallel K_{GWN})$, $M_8 = ID_i' \oplus K'_{GWN-S_j}$, $M_9 = r_g \oplus h(ID_i' \parallel K'_{GWN-S_j})$, $M_{10} = r_g \oplus r_i'$ and $M_{11} = h(ID_i' \parallel SID_j' \parallel K'_{GWN-S_j} \parallel r_i' \parallel r_g)$ and sends $\{M_8, M_9, M_{10}, M_{11}\}$ to S_j.

Step 10. Upon receiving $\{M_8, M_9, M_{10}, M_{11}\}$, S_j computes $ID_i' = M_8 \oplus K_{GWN-S_j}$, $r_g' = h(ID_i'' \parallel K_{GWN-S_j}) \oplus M_9$, $r_i'' = r_g' \oplus M_{10}$, and $M_{11}' = h(ID_i'' \parallel SID_j \parallel K_{GWN-S_j} \parallel r_i'' \parallel r_g')$ and checks if $M_{11}' = M_{11}$. If it does not hold, this session is terminated by S_j; otherwise, S_j generates a random number r_j.

Step 11. S_j computes $M_{12} = r_j \oplus K_{GWN-S_j}$, $SK_j = h(ID_i'' \parallel SID_j \parallel r_i'' \parallel r_g' \parallel r_j)$ and $M_{13} = h(K_{GWN-S_j} \parallel SK_j \parallel r_j)$ and sends the response $\{M_{12}, M_{13}\}$ to GWN.

Step 12. After getting the response $\{M_{12}, M_{13}\}$, GWN computes $r_j' = M_{12} \oplus K'_{GWN-S_j}$ $SK_{GWN} = h(ID_i' \parallel SID_j' \parallel r_i' \parallel r_g \parallel r_j')$ and $M_{13}' = h(K'_{GWN-S_j} \parallel SK_{GWN} \parallel r_j')$ and checks if $M_{13}' = M_{13}$. If it does not hold, this session is terminated; otherwise, this phase proceeds.

Step 13. GWN computes $M_{14} = M_1' \oplus r_g$, $M_{15} = r_i' \oplus r_j'$ and $M_{16} = h(ID_i' \parallel SK_{GWN} \parallel r_g \parallel r_j')$ and sends $\{M_{14}, M_{15}, M_{16}\}$ to U_i.

Step 14. U_i computes $r_g'' = M_{14} \oplus M_1$, $r_j'' = M_{15} \oplus r_i$, $SK_i = h(ID_i \parallel SID_j \parallel r_i \parallel r_g'' \parallel r_j'')$ and $M_{16}' = h(ID_i \parallel SK_i \parallel r_g'' \parallel r_j'')$ and checks if $M_{16}' = M_{16}$. If it does not hold, this session is terminated; otherwise, the authentication process is completed.

After the above, U_i can acquire sensory data from S_j via GWN while a session key SK_i is shared among U_i, S_j and GWN, where $SK_i = SK_j = SK_{GWN}$.

3 Security Analysis of Li *et al.*'s Scheme and Advanced Discussions

In this section, how our found security flaws damage Li *et al.*'s authentication scheme will be shown. First, a legal and malicious user can obtain the secret key K_{GWN-S_j} shared between GWN and S_j after he has acquired sensory data from S_j. After obtaining K_{GWN-S_j}, the legal and malicious user can impersonate S_j to negotiate a session key shared with GWN and the legal user and to deliver forged sensory data. Meanwhile, this malicious user can access S_j without GWN even with a forged identity. Moreover, this user who has successfully obtained K_{GWN-S_j} can reveal the identity of another legal user U_i who also acquires sensory data from S_j, and the innocent user U_i's essential information $h(ID_i \parallel K_{GWN})$ will be retrieved at the same time. Thereupon, the malicious user can impersonate the innocent user U_i to access the desired sensor nodes at will. For clarity and simplicity, we demonstrate how the above security flaws work with U_1 as the malicious user, U_2 as the innocent user and S_1 as the common accessed sensor node. In additional to the found security flaws, further discussions are also made in this section. The details are as follows:

3.1 Leakage of the Secret Key Shared Between GWN and S_j and Impersonating S_j

U_1 is a legal user so he can acquire the sensory data from authorized sensor nodes. In login and authentication phase, U_1 can acquire the sensory data from the specific sensor node S_1 via GWN. It denotes that U_1 is aware of S_1's identity SID_1. U_1 begins to eavesdrop after he sends the login request $\{M_2, M_4, M_5, M_6, M_7\}$ to GWN for acquiring the sensory data from S_1. Within a reasonable period of time, GWN will send $\{M_8, M_9, M_{10}, M_{11}\}$ to S_1, where $M_8 = ID_i' \oplus K'_{GWN-S_j} = ID_1 \oplus K_{GWN-S_1}$, $M_9 = r_g \oplus h(ID_1 \parallel K_{GWN-S_1})$, $M_{10} = r_g \oplus r_i'$ and $M_{11} = h(ID_1 \parallel SID_1 \parallel K_{GWN-S_1} \parallel r_i' \parallel r_g)$. Because U_1 knows his identity ID_1, he can retrieve $K_{GWN-S_1} = M_8 \oplus ID_1$.

However, GWN is responsible for forwarding messages to multiple sensor nodes. It denotes that U_1 may intercept multiple $\{M_8, M_9, M_{10}, M_{11}\}'$s. In this case, U_1 still can reveal K_{GWN-S_1} successfully. To ensure which revealed value is K_{GWN-S_1}, U_1 only needs to do the following.

Step 1. For the intercepted and untested $\{M_8, M_9, M_{10}, M_{11}\}$, U_1 computes $w_1 = M_8 \oplus ID_1$, $w_2 = M_9 \oplus h(ID_1 \parallel w_1)$, $w_3 = M_{10} \oplus w_2$, and $w_4 = h(ID_1 \parallel SID_1 \parallel w_1 \parallel w_3 \parallel w_2)$.

Step 2. U_1 checks if $w_4 = M_{11}$. If it holds, U_1 successfully obtains $K_{GWN-S_1} = w_1$; otherwise, the process will go back to Step 1.

By the above procedure, U_1 can successfully retrieve K_{GWN-S_1} even multiple $\{M_8, M_9, M_{10}, M_{11}\}'$s are intercepted. It is because $w_1 = M_8 \oplus ID_1 = K_{GWN-S_1}$, $w_2 = M_9 \oplus h(ID_1 \parallel w_1) = r_g$, $w_3 = M_{10} \oplus w_2 = r_i$, and $w_4 = h(ID_1 \parallel SID_1 \parallel w_1 \parallel w_3 \parallel w_2) = h(ID_1 \parallel SID_1 \parallel K_{GWN-S_1} \parallel r_i \parallel r_g) = M_{11}$.

On the other hand, U_1 can impersonate S_1 to cheat another legal user U_2 who also wants to access S_1. As shown in the review of Li et al's scheme, U_2 and GWN will execute login and authentication phase when U_2 wants to acquire S_1's sensory data. As a result, GWN will send $\{M_8, M_9, M_{10}, M_{11}\}$ to S_1. Because GWN is responsible for forwarding messages to multiple sensor nodes and S_1's identity is also not revealed in the transmitted $\{M_8, M_9, M_{10}, M_{11}\}$, U_1 eavesdrops and does the following to impersonate S_1.

Step 1. Upon intercepting $\{M_8, M_9, M_{10}, M_{11}\}$, U_1 computes $ID_2'' = M_8 \oplus K_{GWN-S_1}$, $r_g' = h(ID_2'' \parallel K_{GWN-S_1}) \oplus M_9$, $r_i'' = r_g' \oplus M_{10}$, and $M_{11}' = h(ID_2'' \parallel SID_1 \parallel K_{GWN-S_1} \parallel r_i'' \parallel r_g')$.

Step 2. U_1 checks if $M_{11}' = M_{11}$. If it does not hold, this process goes back to Step 1; otherwise, U_1 successfully intercepts the request sent to S_1 and generates a random number r_j.

Step 3. U_1 computes $M_{12} = r_j \oplus K_{GWN-S_1}$, $SK_j = h(ID_2'' \parallel SID_1 \parallel r_i'' \parallel r_g' \parallel r_j)$ and $M_{13} = h(K_{GWN-S_1} \parallel SK_j \parallel r_j)$.

Step 4. U_1 sends the response $\{M_{12}, M_{13}\}$ to GWN.

After getting the response $\{M_{12}, M_{13}\}$, GWN computes $r_j' = M_{12} \oplus K'_{GWN-S_1}$, $SK_{GWN} = h(ID_2' \parallel SID_1' \parallel r_i' \parallel r_g \parallel r_j')$ and $M_{13}' = h(K'_{GWN-S_1} \parallel SK_{GWN} \parallel r_j')$. GWN checks if $M_{13}' = M_{13}$. This must hold, and login and authentication phase proceeds. At last, U_2, GWN, and U_1 will negotiate a shared session key. That is, U_1 can successfully impersonate S_1 and deliver forged sensory data.

3.2 Bypassing GWN

If K_{GWN-S_1} has been revealed by U_1, U_1 can bypass GWN and acquire sensory data from S_1 directly. Moreover, U_1 can acquire S_1's data successfully even with a forged identity. In login and authentication phase, GWN will send $\{M_8, M_9, M_{10}, M_{11}\}$ to S_1, where $M_8 = ID_i' \oplus K'_{GWN-S_1}$, $M_9 = r_g \oplus h(ID_1 \parallel K_{GWN-S_1})$, $M_{10} = r_g \oplus r_i'$ and $M_{11} = h(ID_1 \parallel SID_1 \parallel K_{GWN-S_1} \parallel r_i' \parallel r_g)$. Because U_1 knows K_{GWN-S_1} and SID_1, he can access S_1 without GWN by the following.

Step 1. U_1 generates two random numbers R_1 and R_2.

Step 2. U_1 computes $M_8 = ID_1 \oplus K_{GWN-S_1}$, $M_9 = R_1 \oplus h(ID_1 \parallel K_{GWN-S_1})$, $M_{10} = R_1 \oplus R_2$ and $M_{11} = h(ID_1 \parallel SID_1 \parallel K_{GWN-S_1} \parallel R_2 \parallel R_1)$. Then U_1 sends $\{M_8, M_9, M_{10}, M_{11}\}$ to S_1.

Step 3. After receiving $\{M_8, M_9, M_{10}, M_{11}\}$, S_1 computes $ID_1'' = M_8 \oplus K_{GWN-S_1}$, $r_g' = h(ID_1'' \parallel K_{GWN-S_1}) \oplus M_9 = R_1$, $r_i'' = r_g' \oplus M_{10} = R_2$, and $M_{11}' = h(ID_i'' \parallel SID_j \parallel K_{GWN-S_1} \parallel r_i'' \parallel r_g') = h(ID_1 \parallel SID_1 \parallel K_{GWN-S_1} \parallel R_2 \parallel R_1)$.

Step 4. S_1 checks if $M_{11}' = M_{11}$. It must hold so S_1 generates a random number r_j.

Step 5. S_1 computes $M_{12} = r_j \oplus K_{GWN-S_1}$, $SK_j = h(ID_i'' \parallel SID_j \parallel r_j'' \parallel r_g' \parallel r_j) = h(ID_1 \parallel SID_1 \parallel R_2 \parallel R_1 \parallel r_j)$ and $M_{13} = h(K_{GWN-S_1} \parallel SK_j \parallel r_j)$. Then S_j sends the response $\{M_{12}, M_{13}\}$ to the other communication party. However, the other communication party is U_1 instead of GWN.

Step 6. After getting the response $\{M_{12}, M_{13}\}$, U_1 computes $r_j' = M_{12} \oplus K_{GWN-S_1}$ and $SK_{GWN} = h(ID_i' \parallel SID_j' \parallel r_i' \parallel r_g \parallel r_j') = h(ID_1 \parallel SID_1 \parallel R_2 \parallel R_1 \parallel r_j') = SK_j$.

According to the above, it is ensured that U_1 who has revealed K_{GWN-S_1} can bypass GWN and acquire sensory data from S_1 directly. Furthermore, because S_1 has no information to determine whether the identity of the communication user exists in GWN's database or not, U_1 can use a forged identity to obtain $S1'$s sensory data. In this case, U_1 only needs to execute the above by replacing ID_1 with the forged identity. Meanwhile, S_1 will retrieve the forged identity. Thereupon, even if an audit mechanism is applied, only the forged identity will be traced.

3.3 Revealing Another Legal User′s Identity and Essential Information for Authentication and Impersonating the Innocent User

After U_1 has obtained K_{GWN-S_1}, U_1 can eavesdrop to reveal another legal user U_2's identity and essential information $h(ID_2 \parallel K_{GWN})$ for authentication when U_2 wants to acquire sensory data from S_1. In login and authentication phase, U_2 will send the login request $\{M_2, M_4, M_5, M_6, M_7\}$ to GWN, where $M_2 = sP$, $M_4 = ID_2 \oplus M_3 = ID_2 \oplus sX = ID_2 \oplus sxP$, $M_5 = M_1 \oplus r_i = h(ID_2 \parallel K_{GWN}) \oplus r_i$, $M_6 = h(ID_2 \parallel r_i) \oplus SID_1$ and $M_7 = h(M_1 \parallel SID_1 \parallel M_3 \parallel r_i)$. Upon receiving the login request, GWN checks whether ID_2 exists in the database or not. If ID_2 exists, the phase proceeds and GWN sends $\{M_8, M_9, M_{10}, M_{11}\}$ to S_1, where $M_8 = ID_2 \oplus K_{GWN-S_1}$, $M_9 = r_g \oplus h(ID_2 \parallel K_{GWN-S_1})$, $M_{10} = r_g \oplus r_i$ and $M_{11} = h(ID_2 \parallel SID_1 \parallel K_{GWN-S_1} \parallel r_i \parallel r_g)$.

However, ID_2 and SID_1 are concealed in the transmitted messages, and GWN is responsible for forwarding messages to multiple sensor nodes. It denotes that U_1 may intercept multiple $\{M_2, M_4, M_5, M_6, M_7\}'$s and $\{M_8, M_9, M_{10}, M_{11}\}'$s. U_1 still can reveal ID_2 and $h(ID_2 \parallel K_{GWN})$ successfully. To ensure whether the revealed ID_2 and $h(ID_2 \parallel K_{GWN})$ are correct or not, U_1 only needs to do the following.

Step 1. For the intercepted and untested $\{M_8, M_9, M_{10}, M_{11}\}$, U_1 computes $q_1 = M_8 \oplus K_{GWN-S_1}$, $q_2 = M_9 \oplus h(q_1 \parallel K_{GWN-S_1})$, $q_3 = M_{10} \oplus q_2$, and $q_4 = h(q_1 \parallel SID_1 \parallel K_{GWN-S_1} \parallel q_3 \parallel q_2)$.

Step 2. U_1 checks if $q_4 = M_{11}$. If it holds, it denotes that U_1 has successfully obtained $ID_2 = q_1$, and the procedure proceeds; otherwise, the process will go back to Step 1.

Step 3. Because GWN will send $\{M_8, M_9, M_{10}, M_{11}\}$ to S_1 after receiving $\{M_2, M_4, M_5, M_6, M_7\}$ within a reasonable period of time, U_1 only needs to use $\{M_2, M_4, M_5, M_6, M_7\}'$s received prior to the matched $\{M_8, M_9, M_{10}, M_{11}\}$. For the intercepted and untested $\{M_8, M_9, M_{10}, M_{11}\}$ received prior to the matched $\{M_8, M_9, M_{10}, M_{11}\}$, U_1 computes $q_4 = M_6 \oplus h(q_1 \parallel q_3)$.

Step 4. U_1 checks if $q_4 = SID_1$. If it holds, it denotes U_1 has successfully retrieve U_2's identity ID_2, and U_1 can obtain $h(ID_2 \parallel K_{GWN})$ by computing $h(ID_2 \parallel K_{GWN}) = M_5 \oplus q_3$; otherwise, the process will go back to Step 3.

According to the above, U_1 can retrieve U_2's identity ID_2 and essential parameter $h(ID_2 \parallel K_{GWN})$ for authentication. Thereupon, U_1 can impersonate U_2 because he can compute M_2, M_3, M_4, M_5, M_6, and M_7, send $\{M_2, M_4, M_5, M_6, M_7\}$ to GWN, and compute the shared session key $SK_i = h(ID_2 \parallel SID_1 \parallel r_i \parallel r_g'' \parallel r_j'')$ after receiving $\{M_{14}, M_{15}, M_{16}\}$ from GWN, where $r_g'' = M_{14} \oplus M_1 = M_{14} \oplus h(ID_2 \parallel K_{GWN})$. If different access rights are granted to different users, this security flaw makes a malicious user able to obtain a privileged account.

3.4 Advanced Discussions

As shown in the previous sections, a legal and malicious user can obtain the secret key K_{GWN-S_j} shared between GWN and S_j after he has acquired sensory data from S_j. After obtaining K_{GWN-S_j}, this malicious user can access S_j without GWN even with a forged identity. Moreover, this user who has successfully obtained K_{GWN-S_j} can reveal the identity of another legal user U_i who also acquires sensory data from S_j, and the innocent user U_i's essential information $h(ID_i \parallel K_{GWN})$ will be retrieved at the same time. Thereupon, the malicious user can impersonate the innocent user U_i to access the desired sensor nodes at will.

Why the above security flaws can be successfully mounted in Li *et al.*′s scheme is because of the following reasons. First, a user′s identity is concealed for anonymity. So the secret K_{GWN-S_j} is used to help S_j to retrieve U_i's identity ID_i. However, because U_i is aware of ID_i and K_{GWN-S_j} is constant, U_i can easily retrieve K_{GWN-S_j} from the transmitted parameter $M_8 = ID_i \oplus K_{GWN-S_j}$. Second, only GWN is aware of whether the user communicating with it exists in the system or not, and S_j only can determine whether

$\{M_8, M_9, M_{10}, M_{11}\}$ is sent by GWN because it is supposed that only GWN and S_j know K_{GWN-S_j}. As a result, if K_{GWN-S_j} is compromised, the user who has obtained K_{GWN-S_j} can cheat GWN and S_j. Third, although the concept of key exchange is adopted by U_i and GWN to make only GWN able to retrieve U_i's identity ID_i from $M_4 = ID_i \oplus M_3 = ID_i \oplus sX = ID_i \oplus sxP = ID_i \oplus xsP = ID_i \oplus xM2$, the secret $h(ID_i \parallel K_{GWN})$ can still be retrieved easily. It is because the transmitted parameter $M_5 = M_1 \oplus r_i = h(ID_i \parallel K_{GWN}) \oplus r_i$ and $h(ID_i \parallel K_{GWN})$ is constant. Although r_i's in different sessions should differ from each other, a malicious user who has successfully obtained K_{GWN-S_j} can retrieve r_i and check whether the retrieved $h(ID_i \parallel K_{GWN})$ is correct or not by $M_6 = h(ID_i \parallel r_i) \oplus SID_j$. The possible and feasible strategy to overcome the found security flaws is to combine nonces with K_{GWN-S_j} and $h(ID_i \parallel K_{GWN})$ to make them vary in different sessions.

4 Conclusions

In this paper, we first review a three-factor anonymous authentication scheme for wireless sensor networks in IoT environments. After analyzing their scheme, we find that it suffers from some security flaws. First, a legal and malicious user can obtain the secret key K_{GWN-S_j} shared between GWN and S_j after he has acquired sensory data from S_j. After obtaining K_{GWN-S_j}, the legal and malicious user can impersonate S_j to negotiate a session key shared with GWN and the legal user and to deliver forged sensory data. Meanwhile, this malicious user can access S_j without GWN even with a forged identity. Moreover, this user who has successfully obtained K_{GWN-S_j} can reveal the identity of another legal user U_i who also acquires sensory data from S_j, and the innocent user U_i's essential information $h(ID_i \parallel K_{GWN})$ will be retrieved at the same time. Thereupon, the malicious user can impersonate the innocent user U_i to access the desired sensor nodes at will. If different access rights are granted to different users, this security flaw makes a malicious user able to obtain a privileged account. Why these found security flaws can damage Li *et al.*'s scheme is because nonces are not combined with shared secrets $h(ID_i \parallel K_{GWN})$ and K_{GWN-S_j}. As a result, a malicious user can easily retrieve them. To amend these flaws, different mechanisms to conceal identities and secrets should be adopted.

Acknowledgments

This work was supported in part by Ministry of Science and Technology under the Grant MOST 107-2622-H-025-001-CC3 and in part by National Taichung University of Science and Technology under the Grant NTCUST108-25.

References

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: A survey," *Computer Networks*, vol. 38, no. 4, pp. 393–422, 2002.

[2] R. Amin, S. H. Islam, G. P. Biswas, M. K. Khan, L. Leng, and N. Kumar, "Design of anonymity preserving three-factor authenticated key exchange protocol for wireless sensor network," *Computer Networks*, vol. 101, no. 4, pp. 41–61, 2016.

[3] L. Atzori, A. Iera, and G. Morabito, "The internet of things: A survey," *Computer Networks*, vol. 54, pp. 2787–2805, 2010.

[4] Y. F. Chang, R. K. Huang, and W. L. Tai, "A critique of an anonymity-preserving three-factor authenticated key exchange protocol for wireless sensor networks," in *Proceedings of International Conference on Innovation and Management*, pp. 537–544, Feb. 2017.

[5] D. He, N. Kumar, and N. Chilamkurti, "A secure temporal-credential-based mutual authentication and key agreement scheme with pseudo identity for wireless sensor networks," *Information Sciences*, vol. 321, pp. 263–277, 2015.

[6] X. Huang, X. Chen, J. Li, Y. Xiang, and L. Xu, "Further observations on smart-card-based password-authenticated key agreement in distributed systems," *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 7, pp. 1767–1775, 2014.

[7] Q. Jiang, J. Ma, F. Wei, Y. Tian, J. Shen, and Y. Yang, "An untraceable temporal-credential-based two-factor authentication scheme using ECC for wireless sensor networks," *Journal of Network and Computer Applications*, vol. 76, pp. 37–48, 2016.

[8] X. Li, J. Niu, S. Kumari, F. Wu, A. K. Sangaiah, and K. K. R. Choo, "A three-factor anonymous authentication scheme for wireless sensor networks in internet of things environments," *Journal of Network and Computer Applications*, vol. 103, pp. 194–204, 2018.

[9] S. Ozdemir and Y. Xiao, "Secure data aggregation in wireless sensor networks: A comprehensive overview," *Computer Networks*, vol. 53, no. 12, pp. 2022–2037, 2009.

[10] D. Wang, W. Li, and P. Wang, "Measuring two-factor authentication schemes for real-time data access in industrial wireless sensor networks," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 9, pp. 4081–4092, 2018.

[11] D. Wang and P. Wang, "Two birds with one stone: Two-factor authentication with security beyond conventional bound," *IEEE Transactions on Dependable and Secure Computing*, vol. 15, no. 4, pp. 708–722, 2018.

[12] K. Xue, C. Ma, P. Hong, and R. Ding, "A temporal-credential-based mutual authentication and key agreement scheme for wireless sensor networks," *Journal of Network and Computer Applications*, vol. 36, no. 1, pp. 316–323, 2013.

Biography

Wei-Liang Tai received the Ph.D. degree in computer science and information engineering from National Chung Cheng University, Taiwan, in 2008. He is currently Associate Professor, Department of Information Communications, Chinese Culture University. His main interests are in information security and forensics and multimedia signal processing.

Ya-Fen Chang is a Professor of Department of Computer Science and Information Engineering at National Taichung University of Science and Technology in Taiwan. She received her B.S. degree in computer science and information engineering from National Chiao Tung University and Ph.D. degree in computer science and information engineering from National Chung Cheng University, Taiwan. Her current research interests include electronic commerce, information security, cryptography, mobile communications, image processing, and data hiding.

Po-Lin Hou is a M.S. student of Department of Computer Science and Information Engineering at National Taichung University of Science and Technology in Taiwan. He received the B.I.M. degree in information management from Ling Tung University in Taiwan in 2016. His main interests are in DevOps, information security, and network security.

A Scheme for Finding and Blocking Black Hole Nodes in Mobile Ad Hoc Networks

Zulfiqar Ali Zardari[1], Jingsha He[1,2], Nafei Zhu[1], Muhammad Salman Pathan[1], Muhammad Qasim Memon[3], Muhammad Iftikhar Hussain[1], Peng He[1] and Chengyue Chang[4]

(Corresponding author: Nafei Zhu)

Faculty of Information Technology, Beijing Engineering Research Center for IoT Software and Systems[1]
Beijing University of Technology, Beijing 100124, China
College of Computer and Information Science, China Three Gorges University[2]
Advanced Innovation Center for Future Education, Beijing Normal University[3]
Machinery Industry Information Center, Beijing 100823, China[4]
(Email: znf@bjut.edu.cn)

(Received Dec. 8, 2018; Revised and Accepted May 18, 2019; First Online Spet. 8, 2019)

Abstract

Security of wireless nodes are a major concern in mobile ad hoc networks (MANETs). In this paper, we propose a scheme for detecting and blocking malicious nodes in MANETs. Our proposed scheme relies on specially deployed nodes called FBS nodes to continuously monitor the behaviour of the network nodes. When an FBS node detects a node that exhibits suspicious behaviour, it declares that node as a black hole node. Afterwards, all data and control messages from that node will be discarded. Experiment results show that the proposed scheme can reduce the number of packets dropped by malicious nodes with a low false positive rate.

Keywords: AODV; Blackhole Node; FBS Nodes; Mobile Ad Hoc Network; Throughput

1 Introduction

MANET is a class of wireless networks in which nodes don't rely on a centralised infrastructure to communicate with each other. When nodes lie within the transmission range of each other, they can communicate directly. Otherwise, nodes will rely on the help of other intermediate nodes to realise communication, thus forming a multi-hop communication paradigm. In MANET nodes are independent, can connect anytime and anywhere in the network. Meanwhile, nodes have limited memory, limited energy power, and limited bandwidth. There is no fixed architecture, so nodes act as host well as a router to transfer the data packets to the destination node. Due to the mobility of the nodes, continuous change occurs in the topology. Nodes communicate with each other with different routing protocols to perform the networking function to transmit the data packets from source to destination node [12, 21, 30]. MANET possess various characteristics such as dynamic topology, the absence of central control, shared media, *etc.* MANET can be used in special environments such as emergency operation, battlefields and rescue operations where the rapid deployment of a wired network is difficult.

MANET have various malicious threat such as denial-of-service (DoS) [6, 18, 22]. One type of DoS attacks can be realized in MANETS in which a malicious node provides false routing information during the route finding process to mislead the source node to select an active route to the destination that includes the malicious node so that packet drop behaviour would occur [2, 8, 23]. A DoS attack of this kind is called a black hole attack in which a malicious node drops all the data packets that pass through it. Before data transmission, when a source node wants to send data packets to a destination node, the source node would first check its routing table for any fresh routes heading to destinations node. If no such routes exist, the source node will broadcast an RREQ packet in the network to search for an optimal path. All intermediate nodes, upon receiving the RREQ packet, checks their routing tables for a fresh route to the destination. If no such routes exist, they would further send the RREQ packets to their neighbours that incorporate a hop count number and a destination sequence number in the RREQ packet. Eventually, after receiving the RREQ packet, the destination node sends an RREP packet back to the source node through intermediate nodes that have a fresh route to the destination node.

The choice of an RREP packet by the source node heavily depends on the destination sequence number contained in the packet. If the destination sequence number in an RREP packet is high, the RREP is considered to be optimal. A black hole node always replies to the source node with a very high fabricated destination se-

quence number to distract the source node by choosing the route that includes the black hole node as the fresh path to the destination. Then the black hole node drops all the packets that it gets [4, 10, 20]. Black hole attacks can be launched by a single node or through collaboration. In a single-node black hole attack, only one malicious node is involved, whereas, in a collaborative black hole attack, two or more black hole nodes work in collaboration to disrupt the normal operation of the network. In collaborative black hole attack, two malicious nodes collaborate to commit malicious activities [14, 27, 29].

Many approaches have been proposed in recent years for the detection of collaborative black hole attacks. But few have provided improved results [11, 28]. Although some proposed techniques, such as location-based, trust-based, acknowledgement-based, fuzzy logic and sequence number based, can detect black hole attacks [9, 13, 17, 24, 25], they exhibit the following drawbacks:

- Unable to detect multi-node black hole attacks;
- Unable to detect collaborative black hole attacks;
- High routing overhead and delay;
- Unable to handle the mobility of the nodes.

This paper proposes an improved solution called FBS that can find and isolate collaborative black hole nodes in MANETs. The proposed FBS scheme relies on monitoring nodes that are specially deployed in the network to detect malicious nodes. When a node exhibits any ambiguous behaviour, the monitoring nodes will suspect it to be a malicious node. The, an alert message is broadcast through the network notifying all other nodes about the identity of the malicious node. The proposed FBS scheme has the following advantages and characteristics:

- Monitoring nodes are deployed to cover the entire network and continuously exchange information with each other to cope with the issues of node mobility.
- FBS provides an efficient way of detecting and blocking single and collaborative black hole nodes by incorporating an investigation table.
- FBS improves network performance in the following aspects: Very low false positive rate, high throughput, packet delivery ratio, minimum routing overhead and lower average delay, as compared to existing schemes.
- FBS nodes don't take part in the normal routing process, resulting in higher computation and energy efficiency.

The rest of this paper is organized as follows. Section 2 explains the recent approaches related to the field of interest. Section 3 describes the proposed scheme and explains its functionalities and algorithms. Section 4 presents detailed experimental results. Finally, Sections 5 concludes the paper.

2 Related Works

Jhaveri *et al.* proposed an approach based on the highest threshold value of sequence number to isolate the malicious node [13]. The sequence number based detection scheme (SNBDS) includes three different modes of malicious attacks and have different false routing and selective packet drop attack methods. During routing, the threshold value for the sequence number is calculated at each node. If the difference between the destination sequence numbers in the RREP packet of a particular node is greater than the threshold value of the sequence number, the node is declared as suspicious. A bait request packet (RREQ) with a nonexistent destination address and the destination sequence number is then sent to the suspicious node to confirm its status. If it replies, it is declared as a malicious node. The detection mechanism in this scheme heavily depends upon the bait request. Thus, if the node doesn't respond to the bait request, this mechanism would fail. A lot of control packets are generated in this approach to detect malicious nodes.

Vishvas *et al.* proposed an algorithm to detect a malicious node based on its trust and energy status [16]. The trust value and energy status of a node are used to identify the behaviour of a node. Initially, all the nodes are assigned a trust value of 0.5. A node trust value increases gradually from 0.5 to 1 depending on its packet forwarding behaviour. Energy model calculates the energy of every node in the network as it is assumed that nodes that have high energy values don't participate in the network by not forwarding any kind of data packets further. So, whenever the source node wants to send data packets to the destination node, it only chooses the nodes that have a high value of trust and an energy level that is not greater than a set threshold. The nodes have low trust values, and high energy values are considered as malicious nodes. Due to the dynamic nature of MANETs, some nodes may drop a certain amount of packets because of frequent link breakages, so this mechanism has a high false positive aspect.

Abdelhaq *et al.* proposed the local intrusion detection (LID)-AODV mechanism to find the black hole attack in MANETs [1]. Whenever any node sends RREP packet back to the source node, an intermediate node along the path gets the RREP packet and sends further route request FREQ to the next hop node (NHN) of the sending node. After getting the further route reply FRREP from the NHN node, the intermediate node along the path checks to see whether the NHN node has a valid route to the destination. If it has a valid route, then a node sending the RREP is considered as a normal node. Otherwise, it is a black hole node. The scheme completely fails in the collaborative black hole scenario, where one black hole node behaves as genuine and forwards all the data packets to other collaborative nodes.

Dorri proposed a method to eliminate collaborative malicious nodes from the network by incorporating extended data routing information (EDRI) tables in the

AODV routing protocol [7]. Each EDRI table contains three fields, *i.e.*, FROM (the number of packets received form NHN node), THROUGH (the number of packets sent through the NHN node) and BHN (the black hole node status). Every node maintains the EDRI list to update the status of NHN node. Whenever the source node receives an RREP packet, it checks the EDRI table of each node. If the difference between FROM and THROUGH values exceeds a certain threshold, then the node is considered as a black hole node. A dummy data control packet is further sent to the suspected node to confirm its status. A black hole node will drop the packet, so its status will be confirmed as normal. To avoid false positive rates, every node maintains an extra table to identify the malicious node and sends extra control packet which increases the routing overhead and end to end delay factors.

Kollati *et al.* proposed an algorithm by integrating Integrated Bloom Filter into watchdog algorithm for the detection of the malicious node [15]. A certificate authority (CA) is used in this approach to identify the malicious behaviour of a node by key generation and verification with hashing techniques. During packet forwarding, when a node forwards a data packet, it also embeds a hash value into it. If the hash value between two nodes is not the same, a node is considered as a black hole node as it drops a certain amount of packets. The black hole list is then updated by the CA and the identity of the black hole node is shared in the network so that any kind of transaction is avoided in the future. Extra computation and end to end delay are involved during routing. The obvious mobility conditions can make a node drop some amount of packets, resulting in different hash values, causing a high false positive rate in his approach.

Nissar *et al.* proposed an authentication based scheme to secure MANET using AODV routing protocol against routing attacks [19]. This scheme works in two phases, *i.e.* secure route request phase and secure route reply phase. In the scheme, it is assumed that all the nodes share their public keys with other nodes so that digital certificates can update a repository of nodes. During the first phase, the source node sends the RREQ message in the network by embedding its digital signature. When an intermediate node receives the RREQ packet, it checks the signature in it and, only after verification is successful, does it send further the RREQ by embedding its signature. Otherwise, the RREQ is considered as malicious. During the second phase, when an intermediate node having a fresh route or destination itself sends back RREP, it embeds its private key into the RREP packet. All the intermediate nodes in the reverse path authenticate the key by the sender. If the pattern is correct, it is further sent back to reverse path, else dropped as considered by the malicious node. A high end to end delay is present in this approach

Saluvala *et al.* proposed a technique that provides an authentication mechanism for every node in the network participating during routing in MANETs [3]. Each node in the network, before broadcasting RREQ further, adds 1's complement of its IP address. The receiving node authenticates the RREQ packet of its source by adding the appended one's compliment and source IP address to it to get all ones. For any node not aware of ones compliment of its IP address, all the packets from the node are dropped.

Baqer *et al.* proposed a secure trust-based approach based on the safety status of an RREP packet during routing [5]. Each node in the network maintains two tables, *i.e.*, trust level and malicious node tables. Initially, every node is considered as a trusted node in the network. When an intermediate node in the network receives an RRRP packet, the malicious table is inquired to check if the identity of the RREP packet is already listed as a black hole node. If it is, the RREP packet is dropped. Otherwise, a security procedure is adopted based on the destination sequence number of the node. If the node provides a fabricated destination sequence number that exceeds the threshold value, the trust value of the node is decremented and updated in the list. All the nodes having trust values below one are considered as black hole nodes and included in the malicious node list table.

Thanuja *et al.* proposed a method to avoid collaborative black hole attack by using data routing information (DRI) table [26]. Each node maintains a DRI table to update the information of packet forwarding of its NHN node. When the source node gets an RREP packet from an intermediate node, it checks the DRI table of that node to judge the number of packets is received and then forwarded. If the difference between FROM and 'THROUGH' fields exceeds the set threshold, then the node is considered as a black hole node. Furthermore, a FREQ packet is sent to the malicious nodes NHN to identify the identity of any collaborative black hole node. If the NHN's difference also exceeds the threshold and it doesn't contain any valid route towards the destination, then the nodes are considered as being working in collaboration. An alert packet is then broadcast in the network by the source node with the identities of the malicious nodes.

The proposed FBS scheme is different from the above existing approaches. In this scheme, we don't use beacon messages, bait requests or extra packets to check whether a route is safe. Due to extra routing overhead and delays in the network, some of the existing solutions don't detect collaborative black hole nodes, whereas the FBS scheme can detect collaborative black hole nodes with a very low false positive rate. Moreover, the detection rate of black hole nodes of the proposed scheme is high as compared to the other approaches.

3 The Proposed Finding and Blocking Scheme

The proposed finding and blocking scheme (FBS) relies on the statistical investigation. When a source node wishes to communicate with a destination node, it broadcasts

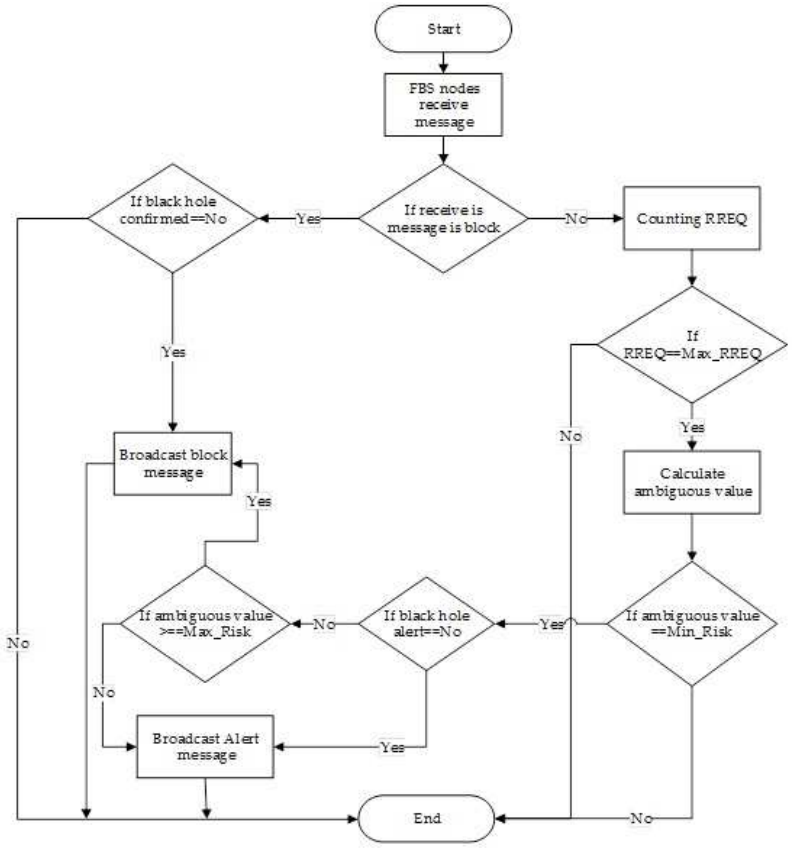


Figure 1: Flowchart of the proposed FBS scheme

RREQ packets in the network. After getting the RREQ packet, the RREP packet is sent back to the source node by an intermediate node having a fresh route to the destination or the destination node itself. In this process, whenever a black hole node gets the RREQ packet, it doesn't forward the RREQ further but sends back the RREP with the highest fake destination sequence number to distract the source node. Whereas in the collaborative attack, some of the RREQ packets are sent by a black hole node to its collaborator. The main intention of black hole nodes not forwarding RREQ is to drop all data packets during the communication process. To cope with this problem, special FBS nodes are employed in FBS which continuously monitors every node in the network in terms of the number of RREQ packets it forwards.

There are two types of nodes in the network. Regular nodes are the normal intermediate nodes which send data to other nodes to exchange the information between each other. Each node maintains a block table in which the identity of the malicious node is saved, broadcast by FBS node as shown in Table 1. All the packets coming from the malicious nodes are then blocked by regular nodes. FBS nodes are the nodes which detect the black hole nodes by some process running on them. Each FBS node maintains an investigation table according to which the decision is made about the status of the node as shown in Table 2.

The node position field describes whether the node is currently in the range of the FBS node. The nodes that move out of the range of the FBS node (i.e., whose RREQs cannot be detected by the FBS node) are set as inactive. According to Table 2, nodes 44 and 32 are active, whereas node 42 is inactive. The RREQ counting field shows that the neighbouring nodes 44, 32, and 42 have broadcast 5, 5 and 6 RREQs, respectively. The Ambiguous Value field in the investigation table represents the current ambiguous value of the respective node as calculated by the FBS node. The Blackhole Alert and Blackhole Confirmed fields show that this or any other FBS node has broadcast an Alert or Block message against the malicious node. The Alert and Block messages are shown in Table 3 and Table 4, respectively. According to the table, only node 42 is declared as a black hole alert, whereas no node is yet announced as a black hole. The primary objective of the FBS nodes is to detect the malicious behaviour of nodes in the network.

The proposed scheme performs the following four tasks:

Maximum request count. At any point when the RREQ count of an individual node reaches the maximum request count, FBS node starts calculating the ambiguous value in the investigation table for each node. Maximum request count is calculated to find any black hole node which is not forwarding the RREQ to neighbour nodes.

Minimum request count. During ambiguous value calculation, if an FBS node finds a node with RREQ

Table 1: FBS investigation table

Node Position	Node ID	RREQ Counting	Ambigious Value	Black Hole Alert	Black Hole Confirmed
Active	44	5	0	No	No
Active	32	5	0	No	No
Inactive	42	6	3	Yes	No

Table 2: Block message

Malevolent Node	FBS Broadcaster Node
51	54

Table 3: Alert message

Malevolent Node	FBS Broadcaster Node
52	53

Table 4: Block table

Malevolent Node ID	FBS Node
51 & 52	54

counts less than minimum request count but positions as active, the ambiguous value id is incremented for that node.

Minimum risk. At any point when the node's ambiguous value is equivalent to minimum risk, an FBS node broadcasts an alert message through the network to notify other FBS nodes. This value is set to half of the maximum risk.

Maximum risk. At any point when the node's ambiguous value is equivalent to maximum risk, an FBS node broadcasts a block message through the network to inform other FBS and regular nodes about the identity of the malicious node.

The following parameters of the FBS node are used for different purposes: Route request count, Ambiguous value, Alert message, and Block message.

Route request count. When an FBS node receives an RREQ packet, it starts counting the RREQs. Each FBS node maintains it's neighbours record in the investigation table. Firstly, it checks whether RREQ sent by a node is already present in the table. If not, it would insert the identity of the fresh entry into the table, and the node position is set as an active node, RREQ Counting to 1, ambiguous value to 0 and black hole alert and black hole confirmed value to NO. If the broadcasting node ID is already in the investigation table, the FBS node will check the black hole confirmed status of the node. If it is yes, then the node is already declared as a black hole, and there is no need to further do any processing. If the black hole confirmed field is NO, then it checks the node position filed in the investigation table and changes it to active if it is inactive. Also, the RREQ count is incremented by one. When the newly obtained value is less than the maximum request count, the process terminates. However, if the value is equal to the maximum request count, the FBS node will start calculating the ambiguous value process. Following is the algorithm for the route request count PROCESS.

Ambiguous value. Ambiguous value is very important in identifying black hole nodes. Ambiguous value process checks all the nodes in the investigation table whose status is active. If the RREQ value of a node is less than the minimum request count, the ambiguous value of that node is incremented. After the increment, if it is equal to minimum risk value and black hole field is NO, the FBS will broadcast an alert message through the network, which includes malicious node ID and its ID, and change the black hole confirmed field of that node to YES. This process continues until all the remaining nodes status is saved in the investigation table. If the new ambiguous value is equal to maximum risk value and black hole confirmation is NO, then a block message is forwarded by the FBS node, which will change the status of black hole confirmation to YES. To reduce the false positive rate (FPR), if a node that has ambiguous value more than zero and acts as a regular node, *i.e.* RREQ forwarded is more than minimum request count, its ambiguous value is decremented, which will decrease the chance of a regular node being declared as a black hole node.

Alert message. When the ambiguous value reaches the minimum risk value, FBS node will send an alert message if it is not previously sent. When this message is received by any regular node, it ignores the message. However, when the FBS node receives an alert message, it finds the malevolent node ID (contained in the alert message) into its investigation table. If the ID is not found, a new entry is created for it, and the ambiguous value is set to the minimum risk value, and the black hole alert is set to YES. The FBS node again broadcasts the alert message. If the investigation table already holds the malevolent node ID, then the black hole alert field is checked. If this field is YES, it indicates that FBS already broadcast the alert message and the process is terminated. If the alert field is NO, then the ambiguous value of the malevolent node is fixed to the minimum risk, and black hole alert is fixed as YES. The purpose of

the alert message is to inform other FBS nodes of the black hole node in the network. Due to the mobility of nodes in MANETs, nodes change their location from one position to another frequently. To cope with this mobility issue, each FBS node shares the information about that node with other FBS nodes, and they are deployed in such a way that they can reach each other. So, if a node changes its location, another FBS node has already had the information of that particular node. Therefore, it can be easily detected wherever it is in the network.

Block message. When the ambiguous value of a node reaches the maximum risk, the FBS node sends a block message (if not previously sent). When a block message is received from FBS node by a regular node, the malevolent ID and broadcaster FBS nodes ID is inserted in the block table by a regular node if they are not added yet.

Meanwhile, when an FBS node receives a block message, it will check the malevolent node ID in its investigation table. If ID doesn't exist in the investigation table, a new entry is inserted in the table and the ambiguous value set to the maximum risk value and black hole alert and confirmation fields set to YES. The FBS node then rebroadcasts the block message. If the malicious node ID is already present in the investigation table, its black hole confirmed field is checked. If the black hole confirmed field value is YES, then it means that the FBS node has already broadcast the block message for that malicious node and the process terminates. If black hole confirmed field is NO, then the malevolent nodes ambiguous value is set to the maximum risk value, black hole alert and confirmation are set to YES and block message is rebroadcast. The purpose of the block message is to inform the normal and FBS nodes in the network of the black hole attack and to spread the message throughout the network with the help of FBS nodes since normal nodes don't participate in broadcasting the block message. The FBS nodes ID is included in the alert message for authentication. Figure 1 is the flow chart of the proposed FBS scheme. When a FBS node receives any message, first it checks to see if the received message is a block message. If it is a block message, the FBS node will check its black hole confirmed value in the investigation table. If not, the FBS node will count RREQ packets of neighbour nodes. If the black hole confirmed value is YES, the FBS node then broadcasts the block message. Otherwise, the process terminates.

Meanwhile, if the RREQ is equal to Max_RREQ, then the FBS node calculates its ambiguous value. If the ambiguous value is equal to minimum risk, then the FBS node checks its black hole alert value. If black hole alert value is "YES", then it broadcasts the alert message if the value is "NO", then it checks its ambiguous value. If the ambiguous value is equal to or greater than the maximum risk, then it broadcasts a block message. Otherwise, it broadcasts an alert message and terminates the process.

Table 5: Simulation parameters

Parameter	**Value**
Network Simulator	NS-2(ver.2.34)
Dimension	1000?1000 m
Regular nodes	200
FBS nodes	9 (fixed)
Mobility model	The random walk mobility model
Simulation time	1000 s
Traffic type	CBR/UDP
Packet size	512 bytes
Mobility speed	0.5-01m/s
Pause time	5-20 s

4 Experiment and Analysis

NS-2 (ver.2.34) was utilized for network simulation to evaluate the performance of the proposed FBS scheme in which 200 regular nodes were deployed in a 1000×1000 m area with nine fixed FBS nodes located in such a way that they can cover all the network area. AODV routing protocol is used in this work. The traffic type constant bit rate (CBR) is used for non-connection oriented traffic model for sending the traffic. The total amount of time for the simulation is 1,000s, the mobility of the nodes varies from 5 to 35 m/s, and the size of the packets is 512 bytes. The transmission range is 250m for all the nodes in the network including the FBS nodes. The proposed FBS scheme is compared to three existing approaches, *i.e.*, the AODV routing protocol, Local Intrusion Detection AODV (LID-AODV) and Hybridization of particle swarm optimization with genetic algorithm (HPSO-GA), to demonstrate the performance of the proposed scheme under black hole attacks. The reason for selecting these approaches are these are similar to the proposed scheme. AODV routing protocol is appropriate for big networks, and it is widely used in literature in the last many years. HPSO-GA neighbor information is used for detection of black hole node, *i.e.* nodes data routing information (DRI). In LID-AODV approach it also takes the information from the previous node and after the next node to detect the malicious node. Whereas in our proposed scheme, special FBS monitoring nodes perform the detection of black hole node by route request RREQ packet from neighboring nodes. Table 5 lists the parameters involved in the simulation.

Detection rate. It is an important metric to show the efficiency of the proposed scheme in detecting malicious nodes. Figure 2 shows the detection rate of different techniques as compared to the proposed FBS scheme. The Figure 2 also shows the overall confidence interval 97% was calculated over ten replications. As can be seen from the Figure 2, the detection rate of the proposed FBS scheme is better than all the participating routing protocols, *i.e.*, 97.33%

at 200 nodes as compared to AODV, LID-AODV and HPSO-GA. In a comparative analysis, it was observed that there are 2.1%, 7.7% and 9.6% improvement in the detection rate of malicious nodes by FSB scheme than HPSO-GA, LID-AODV and AODV, respectively. The reason behind the improved detection rate lies in the efficient detection scheme by FBS nodes. The continuous information sharing among FBS nodes makes them aware of the nodes that frequently move from one position to another due to mobility in MANETs.

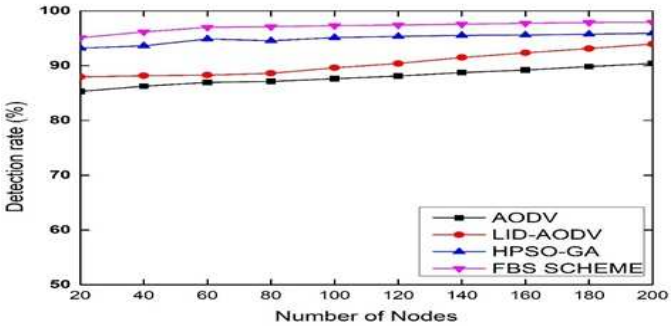


Figure 2: Detection rate (%)

Average delay. Figure 3 shows the average delay concerning time in seconds (sec). As the number of nodes increases in the network, there is an increase in the end to end delay in the network. Because of frequent malicious attacks and mobility conditions require protocols to perform various operations to detect malicious behaviour. A decrease of 0.19s, 0.11s and 0.5s in delay was achieved by the proposed FBS scheme as compared to HPSO-GA, LID AODV and AODV, respectively. Analysis indicates that the FBS scheme has lower end-to-end delay compared to other techniques due to efficient and early detection of black hole nodes in the network. Also, FBS nodes don't participate in normal routing, resulting in less computation in the network.

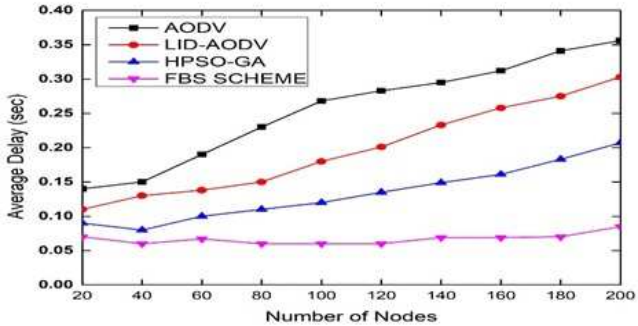


Figure 3: Average delay (%)

Packet delivery rate. Figure 4 shows the packet delivery rate (PDR) of all the participating routing protocols. As the number of nodes increases in the network, there is a drop in the packet delivery rate due to frequent packet drop attacks by malicious nodes. As can be seen from the figure, packet delivery rate of the proposed FBS scheme is higher than other techniques against black hole attacks due to early and efficient detection performed by the FBS nodes during the routing process. PDR of the proposed scheme is 97.45% per 200 nodes, an improvement of 2.49%, 6.93% and 9.21% compared to HPSO-GA, LID-AODV and AODV, respectively. The proposed FBS scheme provides a secure route to destinations with a very low number of malicious nodes during data transmission. Since FBS nodes are effective in the detection of black hole attack by removing malicious nodes from the network quickly, delivery of packets is increased.

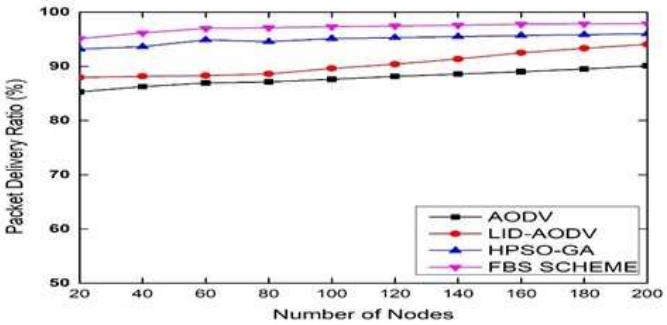


Figure 4: Packet delivery ratio (%)

Throughput. It can be seen from Figure 5 that the throughput of the proposed FBS scheme is much better than other techniques. The throughput of the proposed scheme is 78.9 kbps at 200 nodes, improving the throughput by 5.56%, 9.55%, and 14.92% over HPSO-GA, LID-AODV and AODV, respectively. The reason behind the improved results in the early detection of black hole nodes, which makes the routes to destinations better protected from black hole nodes in turn, increases the throughput by avoiding packet drops.

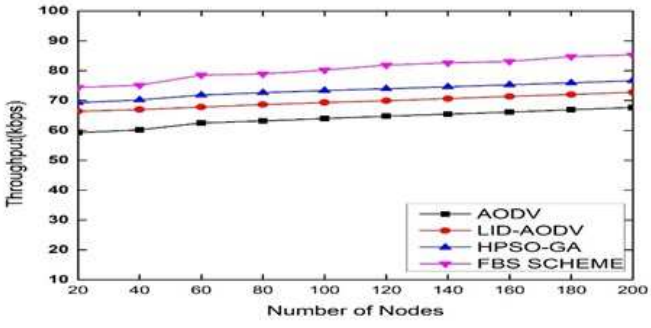


Figure 5: Throughput (%)

Routing overhead. Figure 6 shows the overhead routing comparison of the proposed FBS scheme to other techniques. According to the scenario, the proposed scheme has less routing overhead of 4687 bytes at 200

nodes. Compared to the routing overhead in HPSO-GA, LID-AODV and AODV, the difference is 1300, 1437, 1637 bytes, respectively. At some points, routing overhead is high because the mobility of nodes incurs more control packets to cope with the problem of frequent route hand-offs and new path discoveries.

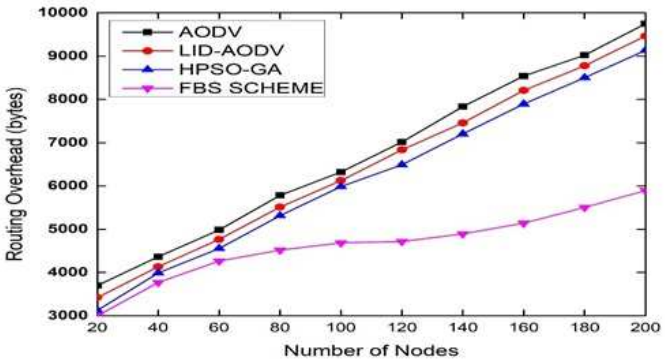


Figure 6: Routing Overhead (%)

5 Conclusion

In this paper, we proposed a solution to deal with black hole nodes in MANETs by using FBS nodes. The objective of the proposed study is to provide both detection and prevention of single and cooperative black hole nodes in the network. In the proposed scheme, FBS nodes continuously monitor the neighbouring nodes in terms of the number of RREQ packets forwarding. Because the black hole nodes don't forwards the RREQ packets to other normal nodes, except few to their collaborators. Whenever an FBS node detects any suspicious behaviour, it increments the ambiguous value for that node. Once the ambiguous value reaches the threshold value, it broadcasts an alert message, and if the ambiguous value exceeds the threshold, it declares the node as a malicious node and broadcast an alert in the network. Finally, all the nodes will add an entry of that node in their block table list and will ignore all the traffic from such node. To manage mobility, all the FBS nodes continuously share the information, to update the information of those nodes which leaves their region.

In addition, the proposed scheme does not require extra processing or any hardware for detection of black hole node. Also, it does not affect the intermediate nodes. The performance of the proposed scheme is compared with some existing techniques, and we found that the proposed scheme provides better outcomes in terms packet delivery ratio, throughput, detection rate, average delay, *i.e.* up to 97.45%, 78.9%, 97.33% and 0.7%, respectively, as compared to other techniques. The FBS nodes provide early and efficient detection of black hole nodes in the network so that the detection rate is high and the packet drop and false positive ratio is very low as compared to the other techniques.

Acknowledgments

The work in this paper has been supported by National Natural Science Foundation of China (61602456).

References

[1] M. Abdelhaq, S. Serhan, R. Alsaqour, and R. Hassan, "A local intrusion detection routing security over manet network," in *Proceedings of International Conference on Electrical Engineering and Informatics*, pp. 1–6, 2011.

[2] S. Aluvala, K. R. Sekhar, and D. Vodnala, "An empirical study of routing attacks in mobile ad-hoc networks," *Procedia Computer Science*, vol. 92, pp. 554–561, 2016.

[3] S. Aluvala, K. R. Sekhar, and D. Vodnala, "A novel technique for node authentication in mobile ad hoc networks," *Perspectives in Science*, vol. 8, pp. 680–682, 2016.

[4] S. K. Arora and H. Monga, "Performance evaluation of manet on the basis of knowledge base algorithm," *Optik*, vol. 127, no. 18, pp. 7283–7291, 2016.

[5] M. Baqer, M. Kamel, I. Alameri, and A. N. Onaizah, "Staodv: A secure and trust based approach to mitigate blackhole attack on aodv based manet," in *IEEE 2nd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC'17)*, pp. 1278–1282,, 2017.

[6] S. Dhende, S. Musale, S. Shirbahadurkar, and A. Najan, "Saodv: Black hole and gray hole attack detection protocol in manets," in *International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET'17)*, pp. 2391–2394, 2017.

[7] A. Dorri, "An edri-based approach for detecting and eliminating cooperative black hole nodes in manet," *Wireless Networks*, vol. 23, no. 6, pp. 1767–1778, 2017.

[8] A. Dorri, S. Vaseghi, and O. Gharib, "Debh: Detecting and eliminating black holes in mobile ad hoc network," *Wireless Networks*, vol. 24, no. 8, pp. 2943–2955, 2018.

[9] A. M. Fahad and R. C. Muniyandi, "Harmony search algorithm to prevent malicious nodes in mobile ad hoc networks (manets)," *Information Technology Journal*, vol. 15, no. 3, pp. 84–90, 2016.

[10] S. Gurung and S. Chauhan, "A dynamic threshold based approach for mitigating black-hole attack in manet," *Wireless Networks*, vol. 24, no. 8, pp. 2957-2971, 2017.

[11] S. Gurung and S. Chauhan, "A review of black-hole attack mitigation techniques and its drawbacks in mobile ad-hoc network," in *International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET'17)*, pp. 2379–2385, 2017.

[12] M. Inam, Z. Li, A. Ali, and A. Zahoor, "A novel protocol for vehicle cluster formation and vehicle head selection in vehicular ad-hoc networks," *International Journal of Electronics and Information Engineering*, vol. 10, no. 2, pp. 103–119, 2019.

[13] R. H. Jhaveri and N. M. Patel, "A sequence number based bait detection scheme to thwart grayhole attack in mobile ad hoc networks," *Wireless Networks*, vol. 21, no. 8, pp. 2781–2798, 2015.

[14] D. Khan and M. Jamil, "Study of detecting and overcoming black hole attacks in manet: A review," in *International Symposium on Wireless Systems and Networks (ISWSN'17)*, pp. 1–4, 2017.

[15] V. K. Kollati, "Ibfwa: Integrated bloom filter in watchdog algorithm for hybrid black hole attack detection in manet," *Information Security Journal: A Global Perspective*, vol. 26, no. 1, pp. 49–60, 2017.

[16] V. H. Kshirsagar, A. M. Kanthe, and D. Simunic, "Trust based detection and elimination of packet drop attack in the mobile ad-hoc networks," *Wireless Personal Communications*, vol. 100, no. 2, pp. 311–320, 2018.

[17] R. Kumar and R. Chadha, "International journal of engineering sciences & research technology mitigation of black hole attack using generic algorithms and fuzzy logic,".

[18] G. Liu, Z. Yan, and W. Pedrycz, "Data collection for attack detection and security measurement in mobile ad hoc networks: A survey," *Journal of Network and Computer Applications*, vol. 105, pp. 105–122, 2018.

[19] N. Nissar, N. Naja, and A. Jamali, "Lightweight authentication-based scheme for aodv in ad-hoc networks," in *International Conference on Wireless Technologies, Embedded and Intelligent Systems (WITS)*, pp. 1–6, 2017.

[20] C. Panos, C. Ntantogian, S. Malliaros, and C. Xenakis, "Analyzing, quantifying, and detecting the blackhole attack in infrastructure-less networks," *Computer Networks*, vol. 113, pp. 94–110, 2017.

[21] M. Pathan, N. Zhu, J. He, Z. Zardari, M. Memon, and M. Hussain, "An efficient trust-based scheme for secure and quality of service routing in manets," *Future Internet*, vol. 10, no. 2, pp. 16, 2018.

[22] M. S. Pathan, J. He, N. Zhu, Z. A. Zardari, M. Q. Memon, and A. Azmat, "An efficient scheme for detection and prevention of black hole attacks in aodv-based manets," *International Journal of Advanced Computer Science and Applications*, vol. 10, no. 1, pp. 243–251, 2019.

[23] A. Rana, D. Sharma, "Mobile ad-hoc clustering using inclusive particle swarm optimization algorithm," *International Journal of Electronics and Information Engineering*, vol. 8, no. 1, pp. 1-8, 2018.

[24] S. Shahabi, M. Ghazvini, and M. Bakhtiarian, "A modified algorithm to improve security and performance of aodv protocol against black hole attack," *Wireless Networks*, vol. 22, no. 5, pp. 1505–1511, 2016.

[25] A. Siddiqua, K. Sridevi, and A. A. K. Mohammed, "Preventing black hole attacks in manets using secure knowledge algorithm," in *International Conference on Signal Processing and Communication Engineering Systems*, pp. 421–425, 2015.

[26] R. Thanuja and A. Umamakeswari, "Black hole detection using evolutionary algorithm for IDS/IPS in manets," *Cluster Computing*, pp. 1–13, 2018.

[27] F. H. Tseng, H. P. Chiang, and H. C. Chao, "Black hole along with other attacks in manets: A survey," *Journal of Information Processing Systems*, vol. 14, no. 1, 2018.

[28] T. Varshney, T. Sharma, and P. Sharma, "Implementation of watchdog protocol with aodv in mobile ad hoc network," in *The Fourth International Conference on Communication Systems and Network Technologies*, pp. 217–221, 2014.

[29] V. S. Venu and D. Avula, "Invincible AODV to detect black hole and gray hole attacks in mobile ad hoc networks," *International Journal of Communication Systems*, vol. 31, no. 6, pp. e3518, 2018.

[30] Z. A. Zardari, J. He, N. Zhu, K. H. Mohammadani, M. S. Pathan, M. I. Hussain, and M. Q. Memon, "A dual attack detection technique to identify black and gray hole attacks using an intrusion detection system and a connected dominating set in manets," *Future Internet*, vol. 11, no. 3, p. 61, 2019.

Biography

Zulfiqar Ali Zardari. 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 is doing PhD in Faculty of Information Technology, Beijing University of Technology, China. He has published six research papers as a first and co-author in national and international journals. His research interest area is Mobile ad hoc networks, Wireless Communications, Information Security, sensor network security, Computer Networks and Network Security.

He Jingsha. He Jingsha received his B.S. degree from Xi'an Jiaotong University in Xi'an, China and his M.S. and PhD degrees from the University of Maryland at College Park in the USA. He is currently a professor in the School of Software Engineering at Beijing University of Technology in China. Professor He has published over 170 research papers in scholarly journals and international conferences and has received nearly 30 patents in the United States and China. His main research interests include information security, network measurement, and wireless ad hoc, mesh and sensor network security.

Nafei Zhu. Nafei Zhu received her B.S. and M.S. degrees from Central South University, China in 2003 and 2006, respectively, and her PhD degree in computer science and technology from Beijing University of Technology in Beijing, China in 2012. From 2015

to 2017, she was a postdoc and an assistant researcher in the Trusted Computing and Information Assurance Laboratory, State Key Laboratory of Computer Science, Institute of Software, Chinese Academy of Sciences in China. She is now on the Faculty of Information Technology at Beijing University of Technology. Dr Zhu has published over 20 research papers in scholarly journals and international conferences (16 of which have been indexed by SCI/EI/ISTP). Her research interests include information security and privacy, wireless communications and network measurement.

Muhammad Salman Pathan. Muhammad Salman Pathan 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 PhD at the Faculty of Information Technology, Beijing University of Technology, China. His research interest is Wireless Communications, Information Security, sensor network security. He has published various research papers in his area.

Muhammad Qasim Memon. Muhammad Qasim Memon received his B.E. degree in Computer System Engineering and M.E. in Information Technology from Mehran University of Engineering & Technology Jamshoro (MUET) Pakistan in 2010 and 2014, respectively. He received his PhD degree in software Engineering from Beijing University of Technology in 2018. Currently, he is working as a Post-Doctoral fellow at advance innovation center for future education (AIFCE), Beijing Normal University, China. His research interests include Text mining, information extraction, text analytics, and wireless sensor networks.

Muhammad Iftikhar Hussain. Muhammad Iftikhar Hussain is currently doing PhD at Faculty of Information Technology, Beijing University of Technology, China. His Research interests include Information Security, Hybrid Cloud Computing Security and Hybrid Cloud Computing Infrastructure and Design. He did his MS Computer Science from Superior University Lahore with distinction. He served as Senior System Engineer in Television and Media Network (Express-News) for four years, one year as System Administrator in 92NEWSHD and two years as Lecturer / Advisor IEEE SUL in Superior University Lahore.

Peng He. Peng He is currently a professor in the College of Computer and Information Technology, China Three Gorges University. He graduated from Hefei University of Technology in 1986 with a bachelor's degree in computer application and from Xi'an Jiaotong University in 1989 with a Master's degree in computer software. He worked in National Time Service Center, Chinese Academy of Sciences (CAS) and participated in 30 research projects, including the seventh national 5-year-plan, the rehearsal of 'eight-five' project from the State Bureau of Surveying and Mapping, CAS youth fund project, Hubei technology research-program, *etc.* Prof. He won the western young scientist's achievement award and the third class award of technology advancement by CAS and the Hubei teaching research achievement award, *etc.* and has been an Education Information Expert of Ministry of Education and a Standing Director of Education Information Technology, Hubei. Prof. He has published over 50 journal papers, some of which have been indexed by EI and ISTP. His research focuses on transmission protocols and information security based on network time synchronization.

Chengyue Chang. Chengyue Chang graduated from Beijing University of Technology in Beijing, China with a Master's degree in software engineering. She is currently an engineer in the Machinery Industry Information Center engaging in the development of Internet software. Ms. Chang's research interests include network security and network forensics.

Android Malware Detection Approaches in Combination with Static and Dynamic Features

Ming-Yang Su, Jer-Yuan Chang, and Kek-Tung Fung
(Corresponding author: Ming-Yang Su)

Department of Computer Science and Information Engineering, Ming Chuan University
5 De Ming Rd., Gui Shan District, Taoyuan City 333, Taiwan
(Email: minysu@mail.mcu.edu.tw)

(Received Sept. 5, 2018; Revised and Accepted Feb. 21, 2019; First Online July 30, 2019)

Abstract

This paper presents two approaches in which static features are combined with dynamic features and used to identify unknown Android malware after proper training by Weka, a well-known machine learning tool. Static features are derived by parsing the test APK after decompilation; they include permissions, sensitive function calls, native-permissions, and priority APK settings. Dynamic features are obtained by parsing an emulator log file after running the test app in the emulator, identifying important activities such as sending short messages (SM) without a user's consent, modifying system files, reading personal contact information, etc. Since the static features can be obtained quickly by parsing the decompiled APK, but dynamic features cannot be obtained in real-time, this study proposes two approaches, ModeA and ModeB, which make efficient use of both types of features. ModeA is a two-tier framework which uses static features for the first tier, and dynamic features for the second tier. Thus, the first tier can be run on a mobile device in real-time, and if the tested app is suspicious then the system can move to the second tier for dynamic feature analysis. ModeB is an off-line system in which static features are merged with dynamic features to measure a test app. ModeB can achieve an overall accuracy of 97.4% for the best case, with ten-fold cross validation in the experiments. The technique of n-fold cross validation, such as n = 2 or 10, is applied to demonstrate the performance of a system for detecting unknown malware.

Keywords: Android Emulator; Android Phones; Dynamic Features; Static Features; Ten-Fold Cross Validation; Weka

1 Introduction

According to the Symantec's annual security report published in March 2018 [25], there were 27K new variants of mobile malware detected in 2017, which was an increase of 54 percent from 17K in 2016. Similarly, a report on malicious mobile software evolution released by Kaspersky in February 2017 [27] showed that about 8.5 million malicious apps were found in 2016, which was 3 times as many as that in 2015. Android smartphones occupy the majority of the market, and are more prone to attacks due to their open source nature. This paper therefore focuses on Android platforms. Android app developers can use either Java, or C/C++ via Android NDK [4] to develop their applications. This convenience has unfortunately resulted in a rapid increase in malware. App developers upload their apps to the official Google Play store, or some unofficial markets, such as Apkpure [5]. Neither, however, has established an effective method of preventing the spread of malware. In addition, numerous free apps contain embedded ad modules for advertising, which makes smartphone security issues even worse. By April 2017, the global accumulative number of Android malware programs reached 19.5 million according to the AV-TEST security report [8]. However, antivirus software normally uses their virus definitions for matching, so when a new malware occurs, the antivirus software may fail to detect it in time. According to the research conducted by Apvrille and Strazzere [7], it takes, on average, three months to spot a new malware in the wild.

Antivirus company Trend Micro selected known malicious apps, classified them, and found that most of them fell into the malware families [26]. This study also showed that the most common behavior of malware is the SMSREG family; all malicious apps belonging to this type carry out malicious activities via Short Message Service (SMS). They steal a user's personal data and send it to a server via short messages, or deliver malware download links via SMS. The second most common malware type was found to be the FAKEINST family, which sends unauthorized short messages to a specific number to register unsolicited, unwanted expensive services for the user, without their consent. Therefore, sending short messages by an app automatically could be regarded as a possible contribution to being classified as malware. In this study,

sending short messages is considered a dynamic feature when analyzing the emulator logs. This study combines static and dynamic app features, and uses Weka to design two malware detection approaches, called ModeA and ModeB, respectively.

Static features are extracted from the app, which are permissions, native-permissions, sensitive function calls and priority-setting values. Dynamic features are obtained by executing the app in a sandbox emulator, and then extracting the activities from the log file of the emulator, such as sending short messages. For ModeA, a two-tier mechanism is proposed in which the first tier employs static features, and the second tier employs dynamic features. For ModeB, all static and dynamic features are merged together to provide a design with higher detection accuracy. Furthermore, a feature weighting strategy is also applied to improve the overall accuracy of ModeB.

The remainder of this paper is arranged as follows. Section 2 offers a literature review, Section 3 describes the framework of ModeA and its related experiment results, Section 4 describes ModeB and its related experiment results, and Section 5 offers some conclusions.

2 Related Researches

Some studies of static, dynamic, or hybrid detections of Android malware are reviewed in this Section. Normally, static analysis relies on features extracted from the app without executing code, while dynamic analysis extracts features based on execution on an emulator.

2.1 Static Detection

Static analysis is a traditional malware detection method for computers, mostly applicable to smart phones. Samra *et al.* [21] used clustering with information retrieval (IR) to identify Android malware. The authors extracted the features of the apps from their XML-files, which declare permissions requested by apps, and then used the Weka K-Mean algorithm for classification. In the paper, the dataset of 18,174 Android apps consisted of 4,612 instances of business and 13,535 instances of tools; the experimental results show that the recall and precision were both 0.71. Liu *et al.* [15] proposed a two-layered permission-based detection scheme for detecting malicious Android apps. The authors considered apps requesting permission pairs as an additional condition, and also considered used permissions to improve detection accuracy.

Zhao and Qian [29] suggested that most malware variants were created by automatic tools, and thus there are special fingerprint features for each malware family. The authors decompiled the Android APK, and mapped the three different kinds of features, Opcodes, API packages and high level risky API functions, to three integrated channels of an RGB image, respectively. They then adopted neural networks to identify each family's features. The experimental results showed that the proposed method successfully identified all 14 malware datasets with an accuracy of 90.67% on average. Fereidooni *et al.* [11] proposed a system called ANASTASIA, which detected a malicious Android app by statically analyzing its behaviors. The authors utilized a large number of statically extracted features from various security behavioral characteristics of an app. A detection framework was built based on machine learning with a high performance detection rate and an acceptable false positive rate. The authors then evaluated the performance on a large-scale malware data-set, including 18,677 malware and 11,187 benign apps, and the results showed a true positive rate of 97.3%, and a false negative rate of 2.7%.

Maier *et al.* [17] also described using obfuscation techniques to bypass static analysis via modifying partial program codes to avoid being similar to known malware samples. They tested and evaluated several antivirus utilities which were able to efficiently identify known malware, but had little success detecting malware after obfuscation. This means that malware detection cannot only rely on static analysis. Some researchers have therefore begun to develop dynamic detection techniques, or combinations of dynamic and static detection techniques.

2.2 Dynamic Detection

Dynamic analysis is based on the behaviors of an application, i.e., the application must be installed and executed in an emulator, and then the log file is analyzed to determine if the application is suspicious. Sun *et al.* [24] indicated that sandbox environments play an important role in the field of information security. A sandbox can execute malware in an isolated environment, minimizing its destructive power, and can test the malware for different ways to find its main intentions. Bhatia and Kaushal [9] presented an approach to perform dynamic analysis of Android apps to classify them as malicious or non-malicious. The authors developed a system which collects and extracts the system call traces of all apps during their runtime interactions with the phone platform. Subsequently all the collected system call data is aggregated and analyzed to detect and classify the behavior of Android applications.

Singh and Hofmann [22] extracted the system call behavior of 216 malicious apps and 278 normal apps to construct a feature vector for training a classifier. The authors applied several classification algorithms to the dataset, including decision tree, random forest, gradient boosting trees, k-NN, Artificial Neural Network, Support Vector Machine and deep learning. Furthermore, three feature ranking techniques, i.e., information gain, Chi-square statistic, and correlation analysis, were used to select appropriate features from the set of 337 system calls. Experiments showed that Support Vector Machines (SVM), after selecting features through correlation analysis, outperformed other techniques, where an accuracy of 97.16% was achieved. Maier *et al.* [18] demonstrate that

Android malware can bypass current automated analysis systems, including AV solutions, mobile sandboxes, and the Google Bouncer. The authors found that malware can either behave benignly or load malicious code dynamically at runtime. They also investigated the frequency of dynamic code loading among benign and malicious apps, and found that malicious apps make use of this technique more often. About one third of 14,885 malware samples were found to dynamically load and execute code, which means traditional antivirus tools can't detect these kinds of malware.

2.3 Hybrid Detection and Other Methods

Qian *et al.* [19] proposed an approach with two steps using static and dynamic analysis separately in each step. The first step used static analysis, and a permission combination matrix was used to determine the risk of the app. For suspicious apps, based on reverse engineering, the authors planted Smali code to monitor sensitive APIs such as sending SMS, accessing user location, device ID, phone number, etc. Their experiment results showed that almost 26% of apps in the Android market have privacy leakage risks. Kapratwar *et al.* [14] suggested that static analysis is more efficient, while dynamic analysis can be more informative, particularly in cases where the code is obfuscated. In this research, the authors applied machine learning techniques to analyze the relative effectiveness of particular static and dynamic features for detecting Android malware.

They also carefully analyzed the robustness of the scoring techniques under consideration. Liu *et al.* [16] decompiled an app to obtain static features by searching the permissions used by the app from the AndroidManifest file, and the APIs from the Smali file. The app was also installed in an emulator to obtain dynamic features from its behaviors. Finally, the static and dynamic vectors were merged into a machine learning system for classification.

Kang *et al.* [13] proposed a method to improve the performance of Android malware detection by incorporating the creator's information as a feature, and classifying malicious applications into similar groups. The proposed system enables fast detection of malware by using creator information such as certificate serial numbers. Additionally, it analyzes malicious behaviors and permissions to increase detection accuracy. The system can also classify malware based on similarity scoring. Its detection rate and accuracy are 98% and 90%, respectively. The MobileSandbox system proposed by Spreitzenbarth *et al.* [23] combines static and dynamic analysis, i.e., the results of static analysis are used to guide dynamic analysis and extend the coverage of executed code. It also uses specific techniques to log calls to native (i.e., "non-Java") APIs, and finally combines these results with machine-learning techniques to classify the analyzed samples as either benign or malicious. Rodriguez-Mota *et al.* [20] proposed a hybrid test framework in which dynamic analysis was implemented after the static analysis of an app. They also analyzed Trojans, and found common features for instances. These features can be used for static analysis to increase classification accuracy.

This study proposes two approaches using static and dynamic features to identify malware on Android platforms. The first, called ModeA, is a two-tier system. ModeA uses static features for the first tier, which can be run in real-time, and uses dynamic features for the second tier. ModeB, is an offline system in which static features are merged with dynamic features to measure a test app. ModeB was able to achieve an overall accuracy of 97.4% for unknown malware in the experiments. The two approaches are introduced in Sections 3 and 4, respectively.

3 System Approach: Mode A - A Two-tier Design

This approach is a two-tier framework, with static and dynamic analysis, operated with an on-line analysis tool, VirusTotal [28], as assistance. The whole structure of this approach is illustrated in Figure 1. It first implements a static analysis in the user's phone, and if the test application is identified as a suspicious app, it is uploaded to the sandbox server for further dynamic analysis. In terms of static analysis, the proposed system extracts the permissions, native-permissions, intent-priority setting, and function calls from the test app. After the app (APK format) is decompiled by Apktool [6], two important files, AndroidManifest.xml and classes.dex, can be obtained. The AndroidManifest.xml contains three kinds of features: permissions, native-permissions and priority settings; the classes.dex contains function calls. Dynamic analysis requires a sandbox server, so that the test app can be uploaded to the server for further analysis. The dynamic analysis uses an Android emulator to run the suspicious app, and the behavior pattern is then extracted to check whether the application contains malicious activities.

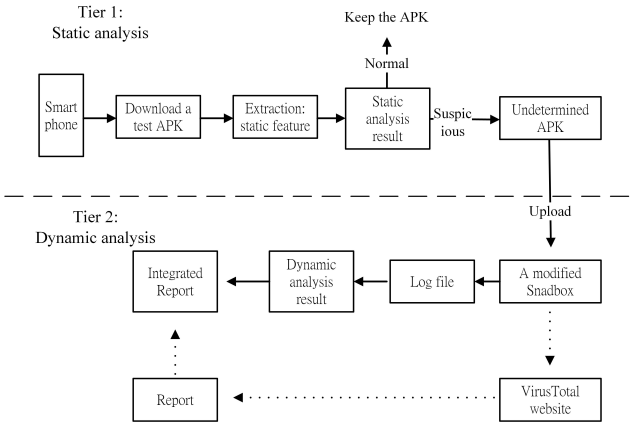


Figure 1: System structure of Mode A: A two-tier design

3.1 Static Analysis

First, the test app is decompiled to obtain static features, which include permissions, native-permissions, intent-priority and sensitive functions. The permission [1] is a security design on Android platforms. If an app wants to execute some specific functions, corresponding permissions must be declared in the AndroidManifest.xml file, and shown to the user before installation so that users are aware of the activities of an app by permission declaration. In this study, the 135 permissions provided by Android 5.0 were considered.

Native-permission is provided by Google to enhance Android programming by supporting other languages, like C/C++, other than JAVA to develop applications. While this flexibility extends Android's supportability, malware developers use this facility to insert malicious codes into applications, or to disguise malicious programs as normal applications, hiding their intentions from the user.

Intent-priority is also declared in Android Manifest.xml, representing the intent-priority of program activity. For example, if the intent-priority value of Application A is larger than that of Application B, related messages are sent to A before they are sent to B. The intent-priority value is preset as 0, and its numerical range is -1000 to 1000. This study found that malware normally sets its intent-priority value higher than normal programs on purpose, so as to make sure the malware receives information first. The value of normal programs usually does not exceed 100.

The final feature of static analysis is sensitive function calls. This study analyzes how many times an application uses sensitive functions as a feature of static analysis. Figure 2 shows partial program codes after a malicious app is decompiled. The malware uses the sendTextMessage( ) function, and sends short messages to a specific number when it is started, and uses the setComponent( ) function to start up another malware. Table 1 lists some of the 59 sensitive functions concerned in this study.

3.2 Dynamic Analysis

In order to carry out dynamic analysis, a sandbox server is built, so that a user can upload an app via mobile phone interface to this server for runtime testing, and then the log file of the emulator can be analyzed to check whether malicious activities are present. This system also uploads the test application to VirusTotal [28] for testing. VirusTotal is a free on-line scanning website. Finally, the system returns the dynamic analysis result and VirusTotal on-line test result to the user. The sandbox environment in this study is an Android virtual machine (Android emulator) [2] provided by Google, which analyzes the activity log in the active stage of applications to identify suspicious activities. MonkeyRunner [3] is an automated testing tool provided by Google, developed based on the Python language. MonkeyRunner is provided via API for developers to write script, after which it sends commands to the Android device to "simulate trigger events". This study prerecords several scripts to simulate user behavior patterns, and MonkeyRunner starts the scripts when an application is running in the Android emulator.

The information exported from an Android system to the log file of the emulator in the preset environment is not specific enough, however. For example, in sending a short message, only the activation of the function is recorded, but the receiving number and the message content are not. In order to detail the information exported to the log file, the emulator's files needed to be modified first in this study. The files comprising the emulator are extracted from the image file "system.img". All functional files are in the .jar format. After decompression, a jar file can generate a classes.dex file, which contains some functional files of the emulator. Then, the de-compilation tool Dex Manager [12] is required, which decompiles a classes.dex file to a smali program code file. After modification, the modified program code (in smali) can then be packaged into classes.dex. Finally, the modified classes.dex is repackaged by compression/decompression tool to the .jar format in order to run the emulator. Take sending a short message (SM) as an example. The function for sending an SM is stored in the SmsManager of telephony-common.jar. The default Android emulator only records the event that the function has been activated, but the content of the message and the recipient of the message are omitted. Modification (see the red frame in the upper part of Figure 3) enables the emulator to record the receiving number and message content in the SendTextMessage file. The yellow frame in the middle of Figure 3 shows that in the log the receiving number and short message content are actually recorded when the SendTextMessage is started. Similarly, Figure 4(a) shows that the short message sending function has been activated, the recipient phone number is 81168, and the content of the message is "SP99". Another example in the log is given in Figure 4(b), where a rename function has been executed, with the names before and after the change. In this study, twelve functions given in Table 2 in the default Android emulator were modified first in order to record more information when they were activated, as if more information about the activities is recorded, then more sophisticated determinations can be performed.

3.3 Dataset and Experimental Results

In this study, normal apps were downloaded, for the most part, from Google Play, and checked by anti-virus software, while malicious apps were mostly obtained from the Contagio Malware [10] website, which periodically updates and shares malware. In total, 900 normal programs and 300 malware programs were applied in the experiments. Thus 1,200 feature vectors representing 1200 apps were obtained and fed into Weka, a machine learning tool, for classification. An instance of one static feature vector is shown in Figure 5. The permissions (black), function calls (brown), and native-permissions (purple) and are set

```
f (paramIntent.getAction().equals("android.intent.action.PACKAGE_ADDED"))

SmsManager.getDefault().sendTextMessage("18670259904", null, " TroJan Ok", null, null);
if (paramIntent.getDataString().substring(8).equals("com.example.com.android.trogoogle"))
{
  Intent localIntent = new Intent();
  localIntent.setComponent(new ComponentName("com.example.com.android.trogoogle",
                                             "com.example.com.android.trogoogle.MainActivity"));
  localIntent.setAction("android.intent.action.VIEW");
  startActivity(localIntent);
```

Figure 2: Illustrations of sensitive functions in applications

Table 1: Some of the 59 Sensitive functions

ContentResolver;->query ;	PackageManager;->installPackage ;
Camera;->open ;	IActivityManager$Stub$Proxy;->shutdown ;
MediaRecorder;->setAudioSource ;	Downloads$ByUri;->startDownloadByUri ;
PowerManager;->reboot ;	Telephony$Mms;->query ;
SmsManager;->sendTextMessage ;	ContentService;->dump;
BluettoothSocket;->connect;	ActivityManagerNative;->restartPackage ;

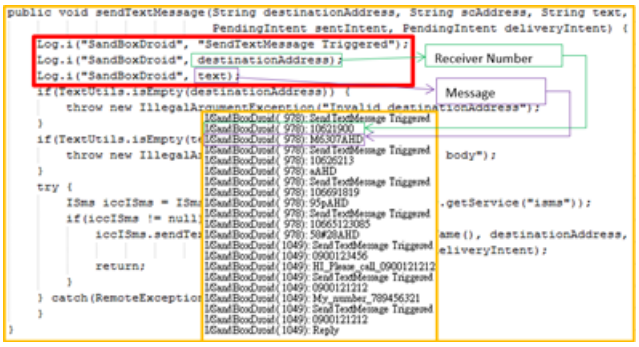
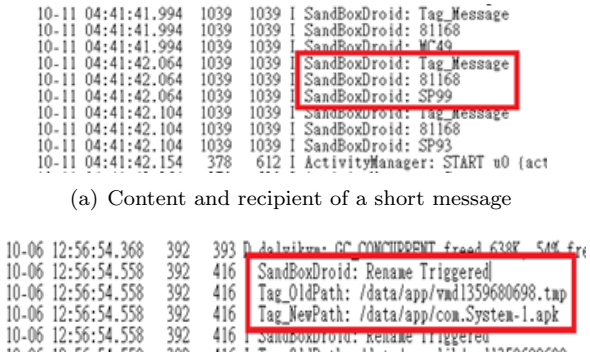


Figure 3: System function modifications



(a) Content and recipient of a short message

(b) The names of a file before and after rename

Figure 4: More information contained in the log of the emulator

Table 2: Modified functions in the Android emulator

1. Turn off background apps	2. Send short messages
3. Execute commands	4. Get GPS data
5. Get device ID	6. Get phone number
7. Turn on camera	8. Delete files
9. Rename files	10. Open files
11. Copy files	12. Retrieve app information

as 1 if they are used, and set as 0 if they are not used. In terms of priority (green), a priority greater than 0 and smaller than 1000 (relatively normal) is set as 0, and one that is less than 0 or greater than 1000 (relatively abnormal) is set as 1. The final feature tells Weka whether the app is malware; "yes" is normal and "no" is malicious.

Table 3 shows the experiment results by Weka. "Non-split" means that all apps are used for training, and tested as well. The n-fold cross-validation (n = 2 or 10) means that all data are divided into n equal parts; n - 1 parts are used for training, and the remainder are used for testing, repeated n times with a different part used each time. Finally, the average value is shown. In terms of accuracy, the best data obtained by non-split, two-fold and ten-fold all occur when the SVM algorithm is used; the values are 93.4%, 89.8% and 91.1%, respectively. Non-split has the maximum value, because the training and test are the same (complete) dataset. The ten-fold experiment yields better data than the two-fold experiment for larger

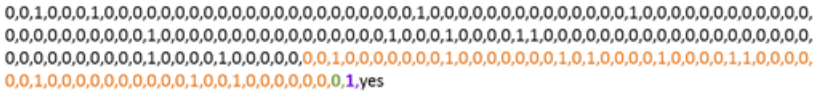


Figure 5: Example of one static feature vector

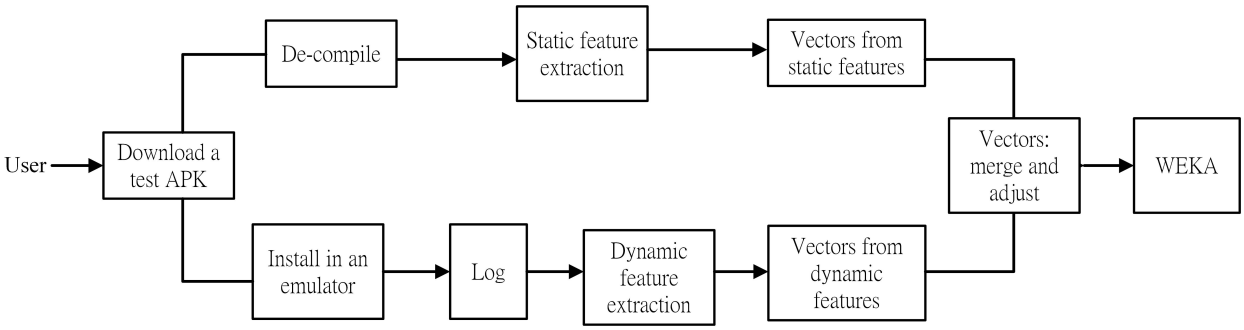


Figure 6: System structure of Mode B: A mixed features design

training sets. Normally, non-split is used to measure the known-attack detection performance of a defense mechanism, and n-fold is used to evaluate its unknown attack detection performance. Like most cases in the fields, both n=2 and n=10 are considered in this paper.

Table 3: Experimental results obtained using static features only

Algorithm \ Result		TP rate	FP rate	Accuracy
Bayes Net	non-split	0.794	0.170	0.803
	two-fold	0.791	0.183	0.798
	ten-fold	0.793	0.167	0.803
Naïve Bayes	non-split	0.781	0.143	0.800
	two-fold	0.787	0.153	0.802
	ten-fold	0.783	0.150	0.800
K-NN	non-split	0.998	0.293	0.925
	two-fold	0.991	0.420	0.888
	ten-fold	0.988	0.380	0.896
J48	non-split	0.997	0.287	0.926
	two-fold	0.983	0.393	0.889
	ten-fold	0.989	0.360	0.902
SVM	non-split	1.000	0.263	0.934
	two-fold	0.970	0.320	0.898
	ten-fold	0.982	0.303	0.911

In terms of dynamic analysis, 30 normal programs and 70 malware programs were selected randomly to form the test set. These programs were installed in the emulator in turn and executed, and MonkeyRunner was started to simulate user behavior. The log was imported into the analysis tool developed in this study, so as to identify suspicious activities. As mentioned in Section 1, sending short messages (SM) without user's awareness is an important feature regarded as malware. In the experiments, the test app was installed in the modified emulator without executing MonkeyRunner [3], *i.e.*, if any SM is sent out, it must be sent by the app without user consent, because no-one is operating the app. Thus far, this the only the criterion for ModeA that has been considered, i.e., if an app sends an SM without user consent, it is regarded as malware. Of the 70 malware programs, 41 applications included such malicious actions. Unreported malware did not trigger the short message sending function, and the 30 normal programs were all identified as having normal behavior. According to the analysis result, as long as an application has an automatic short message sending function, it will be successfully detected in this way. However, malware which executes other attack behaviors (aside from SM sending attacks) cannot be detected. In the following section, approach ModeB merges all static and dynamic features to provide a more precise malware detection design.

4 System Approach: Mode B - All Features Combined, and Feature Weighting

ModeB merges static and dynamic app features, and adjusts their weights appropriately. Similarly, the static features include permissions, native-permissions, functions and priority settings in the application; the dynamic features are obtained by executing the app in the emulator, and then extracting important activities from logs. The overall structure of ModeB is shown in Figure 6. The static features are obtained as in ModeA. Figure 7 shows the tool designed in this study for extracting dynamic features from the log.

After merging the 12 dynamic features with the static features, a feature vector of the app is shown in Figure 8(a), in which the red part, also underlined, represents dynamic features. Table 4 shows the experiment results obtained using Weka. In terms of accuracy, the best experimental results for non-split, two-fold, and ten-fold occurred when the SVM algorithm was used; their values are 95.3%, 92.3% and 93.9%, respectively. Again, non-split obtains the maximum value due to having the same (complete) dataset for training and testing. The ten-fold experiment has better results than the two-fold experiment because of its larger training sets.

Table 4: Experimental results of merging static and dynamic features

Algorithm \ Result		TP rate	FP rate	Accuracy
Bayes Net	non-split	0.787	0.150	0.803
	two-fold	0.790	0.157	0.803
	ten-fold	0.782	0.160	0.797
Naïve Bayes	non-split	0.772	0.143	0.793
	two-fold	0.773	0.143	0.794
	ten-fold	0.776	0.147	0.795
K-NN	non-split	0.998	0.210	0.946
	two-fold	0.992	0.390	0.897
	ten-fold	0.989	0.293	0.918
J48	non-split	0.997	0.227	0.941
	two-fold	0.983	0.317	0.908
	ten-fold	0.989	0.277	0.923
SVM	non-split	1.000	0.190	0.953
	two-fold	0.978	0.243	0.923
	ten-fold	0.991	0.217	0.939

Since there is a large gap between the numbers of static and dynamic features, this study used a weighting method to mitigate the effect resulting from that disparity. The most frequently used feature was given the highest weight, while the other features were compared with the highest feature, and given their corresponding weights. For example, if Feature A has the highest frequency of use, its value is 1000, and the maximum weight is 20 in this paper, so the frequency of use of A is divided by 50 to obtain the weight 20. The other features can be deduced by analogy, divided by 50, and rounded off to obtain their respective weights. The number of dynamic features is quite small in relation to that of the static features, so the maximum weight, i.e. 20, is given to any feature once it occurs. Figure 8(b) shows a feature vector of the app after weight adjustment. The experiment results from Weka are shown in Table 5. The SVM algorithm results in the best accuracy in the three experiments (non-split, two-fold and ten-fold), and their values are 99.5%, 96.3% and 97.4%, respectively. It is clear that the results are improved by adjusting the feature weights. Figure 9 shows the best outcome of the ten-fold experiments using SVM in Weka. The other outcomes of the ten-fold experiments using Bayes Net, Naïve Bayes, K-NN, and J-48 in Weka are given in Figures 10(a), 10(b), 10(c) and 10(d), respectively.

Table 5: Experimental results for weighted features

Algorithm \ Result		TP rate	FP rate	Accuracy
Bayes Net	non-split	0.876	0.180	0.862
	two-folds	0.888	0.187	0.869
	ten-folds	0.878	0.187	0.862
Naïve Bayes	non-split	0.88	0.183	0.864
	two-folds	0.881	0.180	0.866
	ten-folds	0.882	0.183	0.866
K-NN	non-split	0.998	0.02	0.993
	two-folds	0.984	0.187	0.942
	ten-folds	0.981	0.103	0.96
J48	non-split	0.996	0.017	0.993
	two-folds	0.970	0.153	0.939
	ten-folds	0.971	0.110	0.951
SVM	non-split	1.000	0.020	0.995
	two-fold	0.976	0.077	0.963
	ten-fold	0.983	0.053	0.974

Finally, a comparison between the proposed approach and other researches is given below. First, it must be noted that different researches used different methods of showing their performances, with different datasets. There is thus no unanimously fair way to compare them from a specific aspect. The performances of the aforementioned researches in Section 2.3 are summarized because they also used a hybrid approach. The work of [19] was to monitor the information leakage of apps, and the main result is that by experiments, almost 26% applications in the Android market have privacy leakage risks. In [14], the authors used a small dataset with 103 malware and 97 benign apps. They evaluated the performances of static analysis and dynamic analysis separately, instead of merging both types of features together. The authors adopted the term ROC curve, not accuracy, to show the performances of static analysis and dynamic analysis, and

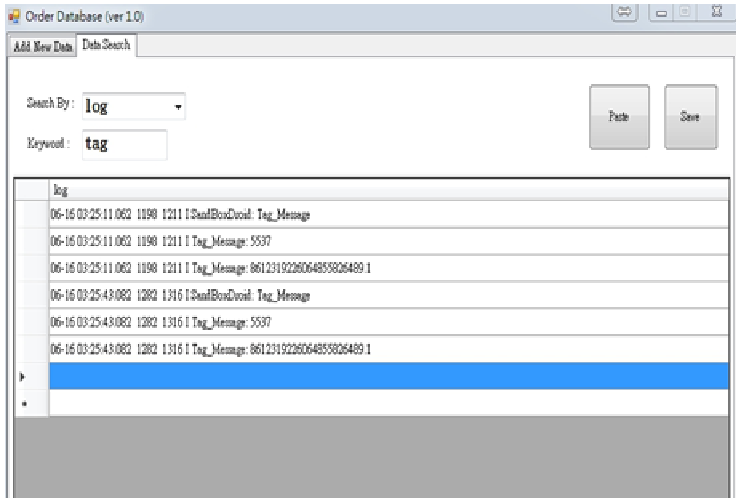


Figure 7: Parsing tool designed to extract dynamic features from the log

0,0,1,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,1,0,0,0,0,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,1,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,1,0,1,0,0,0,0,1,0,0,0,0,1,1,0,0,0,0,
0,0,1,0,0,0,0,0,0,0,0,0,0,0,1,0,0,1,0,0,0,0,0,0,0,0,1,0,0,0,1,0,0,0,0,0,0,1,0,yes

(a) An example feature vector

0,0,7,0,0,0,20,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,6,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,20,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,20,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,6,0,0,0,13,0,0,0,0,4,10,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,17,0,0,0,0,0,0,0,5,0,0,0,0,0,0,0,0,20,0,0,0,0,0,0,0,0,16,0,8,0,0,0,0,0,7,0,0,0,0,0,
9,12,0,0,0,0,0,0,15,0,0,0,0,0,0,0,0,0,0,0,7,0,0,4,0,0,0,0,0,0,0,20,0,0,0,20,0,0,0,0,0,0,20,0,yes

(b) An example feature vector after adjustment

Figure 8: Feature vector for one app after merging static and dynamic features

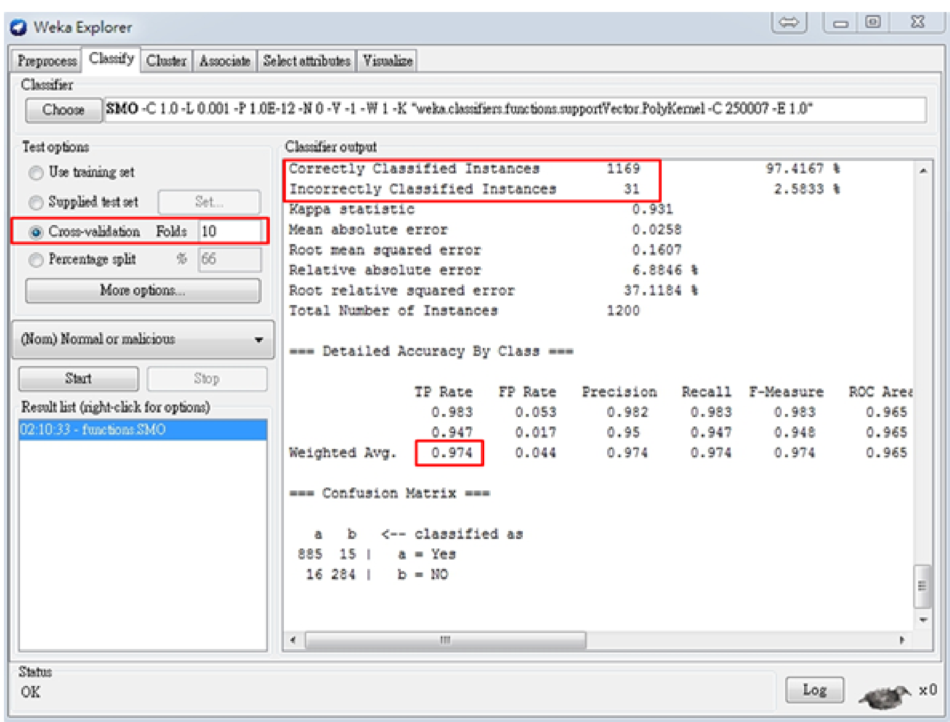
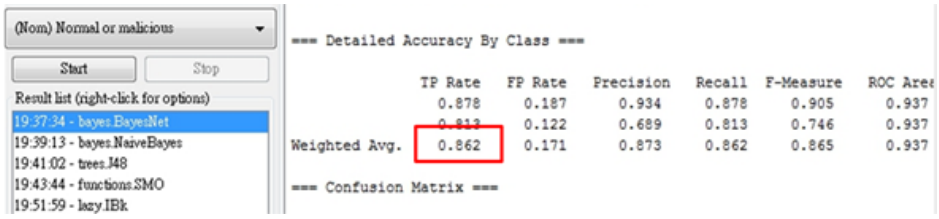
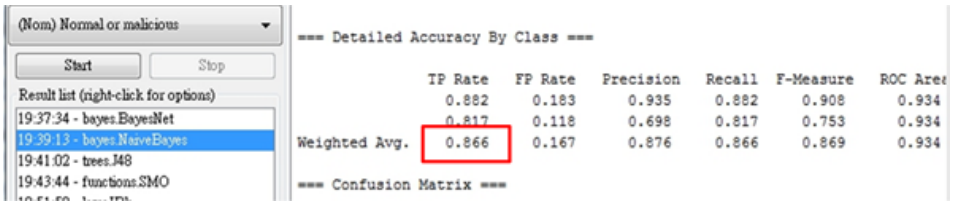


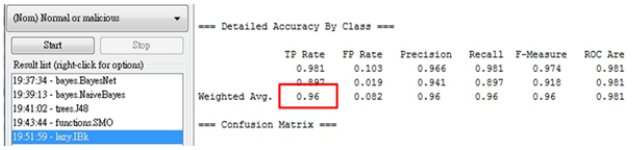
Figure 9: The best outcome of the ten-fold experiments using SVM in Weka



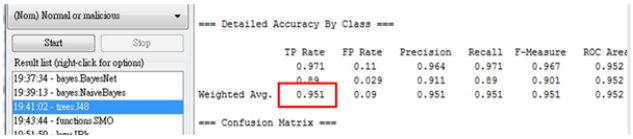
(a) Bayes Net



(b) Naïve Bayes



(c) K-NN



(d) J-48

Figure 10: Outcomes of the ten-fold experiments using different algorithms in Weka

their best results obtained by 10-fold cross-validation were 0.966 and 0.884, respectively, both using the RF100 algorithm. [16] is a short, 2 page paper, in which the authors took 3,414 static features from permissions and API features, and 345 dynamic features from emulator log files. Using a dataset containing 500 malicious apps and 500 benign apps, the authors achieved the best results, in terms of accuracy, for static detection and dynamic detection using SVM (99.28%) and Naïve Bayes (90.00%), respectively. However, the authors did not mention what kind of experimental techniques were adopted, like non-split, 2-fold, 10-fold, etc. In addition, they did not merge static and dynamic features together and resolved to do this in future work.

In [13], the authors proposed an Android malware detection and classification system based on static analysis using serial number information from the certificate as a feature. As a result, the detection system can achieve 98% accuracy, and the classifier module can classify the 20 kinds of malware families with 90% accuracy. The work of [23] did not focus on the classification of malicious and benign apps. Instead, the authors focused on the calls to native (i.e., "non-Java") APIs in apps, because of the potential risks of such calls. They evaluated the system on more than 69,000 apps from Asian third-party mobile markets, and found that about 21% of them actually use native calls in their code. In [20], the authors took 39 trojan Android malware as samples to illustrate the effectiveness of the proposed method. No result regarding classification of a dataset was given.

5 Conclusion

Security mechanisms for Android platforms are fast becoming an important and urgent issue. Current malware technology changes quickly, and malware using obfuscation cannot be identified using only static analysis. Therefore, the malware detection approaches proposed in this paper combine static and dynamic features. Since static features can be obtained by parsing the decompiled test SDK, while dynamic features need to be done with an emulator, two approaches, ModeA and ModeB, are proposed to apply the use of static and dynamic features. ModeA is a two-tier framework in which the first tier can be run on a mobile device to obtain the static features of a test SDK and make a decision in real-time; if necessary, the tested SDK can be uploaded to a server for the second tier check by dynamic features. In ModeB, static features and dynamic features were merged, and since the static features far outnumbered the dynamic features, weights of features were also adjusted to address the disparity. According to the Weka experiments, the overall accuracy values achieved by ModeB for detecting known (non-split) and unknown (ten-fold) malware were 0.995 and 0.974, respectively, both obtained by SVM algorithm.

Acknowledgments

This work was partially supported by the Ministry of Science and Technology, Taiwan, with contracts: MOST 105-2221-E-130-004, MOST 106-2221-E-130-002 and MOST 107-2221-E-130-003.

References

[1] Android, *Android Developer Permission*, June 27, 2019. (https://developer.android.com/reference/android/Manifest.permission.html)

[2] Android, *Android Virtual Device*, June 27, 2019. (https://developer.android.com/studio/run/emulator.html)

[3] Android, *MonkeyRunner*, June 27, 2019. (http://developer.android.com/tools/help/monkeyrunner_concepts.html)

[4] Android, *Android Native Code*, June 22, 2016. (https://developer.android.com/tools/sdk/ndk/index.html)

[5] Apkpure, *Download free Android Games and Android Apps*, June 27, 2019. (https://apkpure.com)

[6] Apktool, *A Tool for Reverse Engineering Android apk Files*, June 27, 2019. (http://ibotpeaches.github.io/Apktool/)

[7] A. Apvrille and T. Strazzere, "Reducing the window of opportunity for Android malware gotta catch'em all," *Journal in Computer Virology*, vol. 8, issue 1-2, pp. 61-71, 2012.

[8] AV-TEST, *Security Report 2016/17*, 2017. (https://www.av-test.org/fileadmin/pdf/security_report/AV-TEST_Security_Report_2016-2017.pdf)

[9] T. Bhatia and R. Kaushal, "Malware detection in android based on dynamic analysis," in *International Conference on Cyber Security and Protection of Digital Services (Cyber Security)*, pp. 1-6, 2017.

[10] Contagio mobile, *Android Fakebank Samples*, Mar. 20, 2018. (http://contagiominidump.blogspot.com/)

[11] H. Fereidooni, M. Conti, D. Yao, and A. Sperduti, "ANASTASIA: android malware detection using static analysis of applications," in *8th IFIP International Conference on New Technologies, Mobility and Security (NTMS'16)*, pp. 1-5, 2016.

[12] Jasi2169_TeamURET, *Dex Manager v1.1 - Designed To Play With Classes.dex*, Dec. 30, 2014. (http://forum.xda-developers.com/android/software/tool-dex-manager-v1-0-designed-to-play-t2988532)

[13] H. Kang, J. W. Jang, A. Mohaisen, and H. K. Kim, "Detecting and classifying android malware using static analysis along with creator information," *International Journal of Distributed Sensor Networks*, online published in Jan. 2015.

[14] A. Kapratwar, F. D. Troia, and M. Stamp, "Static and dynamic analysis of android malware," in *3rd International Conference on Information Systems Security and Privacy (ICISSP'17)*, pp. 653-662, 2017.

[15] X. Liu and J. Liu, "A two-layered permission-based android malware detection scheme," in *IEEE International Conference on Mobile Cloud Computing, Services and Engineering*, pp. 142-148, 2014.

[16] Y. Liu, Y. Zhang, H. Li, and X. Chen, "A hybrid malware detecting scheme for mobile android applications," in *IEEE International Conference on Consumer Electronics (ICCE'16)*, pp. 155-156, 2016.

[17] D. Maier, T. Muller, and M. Protsenko, "Divide-and-conquer: why android malware cannot be stopped," in *9th International Conference on Availability, Reliability and Security (ARES'14)*, pp. 30-39, 2014.

[18] D. Maier, M. Protsenko, and T. Muller, "A game of Droid and Mouse: The threat of split-personality malware on Android," *Computers & Security*, vol. 54, pp. 2-15, Oct. 2015.

[19] Q. Qian, J. Cai, M. Xie, R. Zhang, "Malicious behavior analysis for android applications," *International Journal of Network Security*, vol. 18, no.1, pp.182-192, 2016.

[20] A. Rodriguez-Mota, P. J. Escamilla-Ambrosiot, S. Morales-Ortega, M. Salinas-Rosale, and E. Aguirre-Anaya, "Towards a 2-hybrid android malware detection test framework," in *IEEE International Conference on Electronics, Communications and Computers*, 2016.

[21] A. A. A. Samra, Y. Kangbin, and O. A. Ghanem, "Analysis of clustering technique in android malware detection," in *International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS'13)*, pp. 729-733, 2013.

[22] L. Singh and M. Hofmann, "Dynamic behavior analysis of android applications for malware detection," in *International Conference on Intelligent Communication and Computational Techniques (ICCT'17)*, pp. 1-7, 2017.

[23] M. Spreitzenbarth, T. Schreck, F. Echtler, D. Arp, and J. Hoffmann, "Mobile-Sandbox: combining static and dynamic analysis with machine-learning techniques," *International Journal of Information Security*, vol. 14, no. 2, pp 141–153, 2015.

[24] L. Sun, S. Huang, Y. Wang, and M. Huo, "Application policy security mechanisms of Android system," in *14th IEEE International Conference on High Performance Computing and Communications*, pp. 1722-1725, 2012.

[25] Symantec, *Internet Security Threat Report*, 2018. (https://www.symantec.com/security-center/threat-report)

[26] Tread, *Masque, FakeID, and Other Notable Mobile Threats of 2H 2014*, Jan. 7, 2015. (https://www.trendmicro.com/vinfo/us/security/news/mobile-safety/masque-fakeid-and-other-notable-mobile-threats-of-2h-2014?linkId=11679913)

[27] R. Unuchek, *Mobile Malware Evolution 2016*, Feb. 28, 2017. (https://securelist.com/analysis/kaspersky-security-bulletin/77681/mobile-malware-evolution-2016/)

[28] VirusTotal, *Analyze Suspicious Files and URLs to Detect Types of Malware Including Viruses, Worms, and Trojans*, June 27, 2019. (https://www.virustotal.com)

[29] Y. L Zhao1 and Q. Qian, "Android malware identification through visual exploration of disassembly files," *International Journal of Network Security*, vol.20, no.6, pp.1061-1073, 2018.

Biography

Ming-Yang Su received his B.S. degree from the Department of Computer Science and Information Engineering of Tunghai University, Taiwan in 1989, and received his M.S. and Ph.D. degrees from the same department of the National Central University and National Taiwan University in 1991 and 1997, respectively. He is an IEEE member, and currently a professor of the Department of Computer Science and Information Engineering at the Ming Chuan University, Taoyuan, Taiwan. His research interests include network security, intrusion detection/prevention, malware detection, mobile ad hoc networks, Mobile security and wireless sensor networks.

Jer-Yuan Chang received the M.S. degree in 2016, from the Department of Computer Science and Information Engineering of Ming Chuan University, Taoyuan, Taiwan. His research interests are in the areas of MANET security, Mobile security, intrusion detection and prevention.

Kek-Tung Fung received the M.S. degree in 2014, from the Department of Computer Science and Information Engineering of Ming Chuan University, Taoyuan, Taiwan. His research interests are in the areas of Mobile security and SIP security.

Probabilistic RSA with Homomorphism and Its Applications

Yaling Geng, Shundong Li, and Sufang Zhou
(Corresponding author: Shundong Li)

School of Computer Science, Shaanxi Normal University, Xi'an 710062, China
(Email: gengtheory@snnu.edu.cn)

(Received Mar. 15, 2018; Revised and Accepted July 13, 2018; First Online Jan. 26, 2019)

Abstract

RSA is the most famous and the most efficient public key encryption algorithm. However it is deterministic and this seriously restricts its applications in designing general cryptographic protocols. In this paper, we modify the RSA scheme to obtain two probabilistic encryption algorithms that are semantically secure and multiplicatively homomorphic. The two probabilistic RSA variants with higher security can be used either to encrypt or to sign messages, and they can also resist homomorphic attacks. Furthermore, the improved RSA variant algorithms can be extensively applied in designing various cryptographic protocols, such as digital commitment protocol, zero-knowledge proof protocol, oblivious transfer protocol and secure multi-party computation protocol. They provide new efficient tools for cryptographic protocol designing. Theoretical analysis and implementations show that the improved RSA variants are secure and efficient.

Keywords: Homomorphism; Homomorphism Attack; Public Key Encryption; Probabilistic Encryption; Probabilistic Signature

1 Introduction

RSA is the first public key encryption algorithm [44]. It can be used not only for encryption but also for digital signature, and it is simple and easy to implement. For a long time, the researchers continue to improve the efficiency of RSA by using a variety of hardware and software technologies [5, 27, 28], which makes RSA become one of the most efficient public key encryption algorithms and the international standard of asymmetric encryption system. In addition, RSA is also widely used in key distribution, public key encryption, digital signature, authentication and other fields, and it becomes the standard in these fields. Moreover, RSA occupies an extremely important position in many aspects of cryptography and information security.

However, RSA is a deterministic public key encryption algorithm, which means it could not resist chosen plaintext attack, that is, attackers can randomly select a certain number of plaintexts, and encrypt the plaintexts to obtain the corresponding ciphertexts. After comparing the corresponding ciphertexts with intercepted ciphertexts, the attackers determine whether the corresponding paintexts are the chosen plaintexts. Chosen plaintext attack is very effective if plaintext space is small. RSA cannot resist chosen plaintext attack because it is deterministic, which largely limits its application to other aspects of cryptography.

In order to make public key encryption algorithm resist chosen plaintext attack, Goldwasser and Micali proposed a solution, namely probabilistic encryption scheme. The scheme proposed the concept of probabilistic encryption for the first time [18,19], and it has been widely accepted later. The new cryptosystem is defined as probabilistic encryption cryptosystem (PEC). PEC is a kind of non-deterministic public key cryptosystem, and the essence of which is to add random parameters in the encryption, such that the corresponding ciphertexts of two same plaintexts are completely different even using the same encryption key. In brief, there is one-to-many correspondence between plaintexts and the corresponding ciphertexts in PEC.

Probabilistic encryption algorithms, such as the ElGamal [10], the Pailliar [39], the Okamoto-Uchiyama [38], elliptic curve cryptography [25,36] and NTRU [24] can be widely used in new fields of cryptography such as oblivious transfer [42], zero-knowledge proof [20], bit commitment [37], secret sharing [47], or secure multi-party computation [17]. Deterministic RSA encryption algorithm cannot be used to construct bit commitment protocols, oblivious transfer protocols and zero-knowledge proof protocols.

In the basic public key encryption systems, RSA and the Rabin [43] are deterministic encryption algorithms with homomorphism, and the ElGamal, the Paillier, the Okamoto-Uchiyama, NTRU and elliptic curve are probabilistic encryption algorithms with homomorphism. Uncertainty makes that public key encryption algorithm

has higher security and can resist chosen plaintext attack. A more important reason that above probabilistic encryption algorithms are widely applied is that they are homomorphic. Homomorphism makes these algorithms be widely used in secure multi-party computation [7,23,29–31,33,41,49,50]. Because of the advantage of the importance and efficiency of RSA, it is of very important theoretical significances and practical significances to modify RSA to probabilistic encryption algorithm with homomorphism, which will greatly extend the range of applications of RSA, such that RSA will play a bigger role in cryptography and information security practice.

So far, the improvement schemes of RSA algorithm are divided into three categories as follows.

1) The improved algorithms are probabilistic without homomorphism [2, 14, 48, 52];

2) The improved algorithms are neither probabilistic nor homomorphic [11, 34];

3) The improved algorithms are probabilistic with homomorphism [8].

Bellare *et al.* [2] first proposed randomized filling technique and the corresponding optimal asymmetric encryption filling scheme, namely RSA-OAEP. Shoup *et al.* [48] and Fujisaki *et al.* [14] both presented the improvement schemes of RSA-OAEP. These randomized filling solutions guarantee that the probabilistic RSA that are modified from deterministic RSA is more secure in the random oracle model. The probabilistic RSA not only improves efficiency of encryption but also extends application range. However, it is not homomorphic, which limits its applications in the fields of secure multi-party computation and so on, and the length of plaintexts have decreased. In addition, the security of the scheme is based on random oracle model, but the security of random oracle model is based on an ideal hash function and the ideal hash function does not actually prove existing. Therefore the security of the scheme remains to be studied.

Yu *et al.* [52] modified RSA to probabilistic encryption algorithm by bringing in random numbers and improved the efficiency. However, the new algorithm also loss the homomorphism, which results in that the new algorithm cannot be widely used in cryptography and information security. Makkaoui *et al.* [11] described an improved RSA encryption scheme, namely "Cloud-RSA". The new scheme is able to resist many known brute force attacks and to maintain multiplicative homomorphism, but it does not guarantee the confidentiality of a key exchange and it is a deterministic encryption scheme, which cannot be applied to bit commitment and digital signature.

Liu *et al.* [34] constructed an improved RSA algorithm using two combinatorial identities based on RSA public key encryption algorithm. It can partly resist common modulus attack, but it is a deterministic encryption algorithm, which cannot resist chosen plaintext attack. Dhakar *et al.* [8] designed a modified RSA encryption algorithm with additive homomorphism, but its execution time is almost 6 times of RSA algorithm because it needs more modulus exponentiations.

In order to take full advantages of simple principle and low computational complexity of RSA, and to enable RSA algorithm to play an important role in oblivious transfer, zero-knowledge proof, bit commitment, secret sharing and secure multi-party computation, this paper proposes two secure and efficient RSA public key encryption variants. These schemes keep multiplicative homomorphism and have semantic security, and therefore can be widely applied to above new fields of cryptography. The improvement schemes not only have important theoretical significances in cryptography but also have important practical significances in constructing other cryptographic encryption protocols, and they have broad applications in privacy protection, secure multi-party computation, cloud storage and cloud computing.

Our contributions: The main contributions of this study are as follows.

1) We design two secure and efficient probabilistic RSA variants with homomorphism using theory of number and public key encryption algorithm, and prove that they are correct and semantically secure. Homomorphism and nondeterminacy will make the algorithms be able to be widely used in all kinds of new cryptography fields;
2) We present a probabilistic digital signature scheme and a digital commitment scheme based on one improved algorithm, and expand the scope of research fields and practical applications of public key cryptography.

Paper organization: The remainder of the paper is organized as follows: Section 2 describes some preliminaries. Section 3 proposes an improved probabilistic RSA algorithm and proves its correctness, multiplicative homomorphism and security. Section 4 presents an efficient probabilistic RSA algorithm with multiplicative homomorphism and proves that the algorithm is secure and correct. Section 5 presents concrete application examples of the improved algorithm. Section 6 analyzes the efficiency and illustrates the results of experiments. Section 7 concludes our work with possible further research directions.

2 Preliminaries

In this section, we introduce several basic knowlege about RSA.

2.1 Public Key System

Diffie and Hellman proposed public key cryptosystem in 1976 [9], which is also known as asymmetric cryptosys-

tem. The most important characteristic of public key cryptosystem is as follows: the keys exist in pairs and it is intractable to compute one key from another key. One is called public key, and the other is called private key. Messages encrypted with public key can only be decrypted by using the corresponding private key in public key cryptosystem. A traditional public key cryptography algorithm usually consists of three algorithms [15]: $\text{KeyGen}_{\mathcal{E}}$, $\text{Encrypt}_{\mathcal{E}}$, and $\text{Decrypt}_{\mathcal{E}}$.

$\text{KeyGen}_{\mathcal{E}}$. Taking a security parameter λ as the input(The λ is the bits of large prime numbers), $\text{KeyGen}_{\mathcal{E}}$ outputs a private key sk, a public key pk and the corresponding plaintext space $\mathcal{P}$ and ciphertext space $\mathcal{C}$.

$$(skpk\mathcal{PC}) \leftarrow \text{KeyGen}_{\mathcal{E}}(\lambda).$$

$\text{Encrypt}_{\mathcal{E}}$. Taking the public key pk and a plaintext $M \in \mathcal{P}$ as inputs, $\text{Encrypt}_{\mathcal{E}}$ outputs the corresponding ciphertext $C \in \mathcal{C}$.

$$(C) \leftarrow \text{Encrypt}_{\mathcal{E}}(pk, M)(M \in \mathcal{P}).$$

$\text{Decrypt}_{\mathcal{E}}$. Taking the private key sk and a ciphertext $C \in \mathcal{C}$ as inputs, $\text{Decrypt}_{\mathcal{E}}$ outputs the plaintext $M \in \mathcal{P}$.

$$(M) \leftarrow \text{Decrypt}_{\mathcal{E}}(sk, C)(C \in \mathcal{C}).$$

2.2 Homomorphic Encryption

Homomorphic encryption algorithm plays a very important role in secure multi-party computation. Homomorphism is the most important property of the ElGamal, the Paillier, the Okamoto-Uchiyama, NTRU and elliptic curve public key encryption algorithm, which makes these algorithms be powerful building blocks in constructing other cryptographic protocols. A homomorphic encryption algorithm $\mathcal{E}$ consists of algorithms $\text{KeyGen}_{\mathcal{E}}$, $\text{Encrypt}_{\mathcal{E}}$, $\text{Decrypt}_{\mathcal{E}}$ and $\text{Evaluate}_{\mathcal{E}}$, which inputs the public key pk, the operation S and ciphertext group $\mathbb{C} =< C_1, \cdots, C_l >$, and outputs the ciphertext of $S(M_1, \cdots, M_l)$.

$$\text{Encrypt}_{\mathcal{E}}(pk, S(M_1, \cdots, M_l)) \leftarrow \text{Evaluate}_{\mathcal{E}}(pk, S, \mathbb{C}).$$

2.3 RSA Public Key Cryptosystem

Rivest, Shamir and Adleman proposed the famous RSA public key cryptosystem in 1978. Its security is based on the large integer factorization problem. So far, it is the most mature public key encryption algorithm in cryptography.

KeyGen.

1) Choose two large prime numbers p and q;
2) Compute $n = p \times q$ and $\varphi(n) = (p-1)(q-1)$, where $\varphi(n)$ is the value of Euler toient function of n;
3) Choose an integer e such that $1 < e < \varphi(n)$ and $\gcd(\varphi(n), e) = 1$;
4) Compute d such that $d \times e \equiv 1 \bmod \varphi(n)$;
5) The public key is (e, n), and the private key is (d, n).

Encrypt. To encrypt message m, compute

$$c \equiv m^e \bmod n.$$

Decrypt. To decrypt ciphertext c, compute

$$m \equiv c^d \bmod n.$$

Specifically, RSA is multiplicatively homomorphic, that is,

$$\begin{aligned} E(M_1) \times E(M_2) \bmod n &= ({M_1}^e \bmod n) \times ({M_2}^e \bmod n) \\ &= (M_1 \times M_2)^e \bmod n \\ &= E(M_1 \times M_2) \bmod n. \end{aligned}$$

2.4 RSA Blinding

RSA blinding is usually used to sign a message. The details are as follows.

Blind RSA signature [17]: the author of the message computes the product of the message and blinding factor, i.e.:

$$m' = mr^e \bmod n$$

and sends m' to the signer. The signer then computes the blinded signature s' as:

$$s' = (m')^d \bmod n.$$

s' is sent back to the author of the message, who can then remove the blinding factor to reveal s, the valid RSA signature of m:

$$s = s'(r)^{-1} \bmod n.$$

This works because RSA keys satisfy the equation $r^{ed} \equiv r \bmod n$ and thus

$$s = s'(r)^{-1} = (m')^d(r)^{-1} = m^d r^{ed}(r)^{-1} = m^d \bmod n.$$

Hence s is indeed the signature of m.

This process clearly shows that who adds the blind factor can remove it. This property restricts its application in secure a communication where the sender can add a blind factor, but the receiver cannot remove it. RSA blinding attack may trick the signer into decrypting a message by blind signing another message [12]. Since the signing process is equivalent to decrypting with the signer's secret key, an attacker can provide a blinded version of a message m encrypted with the signer's public key, m' for them to sign. The encrypted message would usually be some secret information which the attacker observed being sent encrypted under the signer's public key which the attacker wants to learn more about. When the

attacker removes the blindness of the signed version they will have the clear text:

$$\begin{aligned} m'' = m'r^e \bmod n &= (m^e(\bmod n) \cdot r^e)(\bmod n) \\ &= (mr)^e \bmod n. \end{aligned}$$

where m' is the encrypted version of the message. When the message is signed, the cleartext m is easily extracted:

$$\begin{aligned} s' = (m'')^d \bmod n &= ((mr)^e \bmod n)^d \bmod n \\ &= mr^{ed} \bmod n \\ &= m \cdot r \bmod n, \end{aligned}$$

since

$$ed = 1 \bmod \varphi(n).$$

Note that $\varphi(n)$ refers to Euler's totient function. The message is now easily obtained.

$$m = s' \cdot r^{-1} \bmod n = mr \cdot r^{-1} \bmod n = m \bmod n.$$

This attack works not only for signing the result of a cryptographic hash function applied to the message but also for signing the message itself.

2.5 Security

Semantic security [6,53] is an important index to measure the security of public key cryptosystem, and it means that an adversary cannot obtain any message about plaintexts. Generally, semantic security of an encryption scheme is characterized by an indistinguishable game, which is also called IND game. IND game is a kind of mental experiment, which has two participants. One is called challenger($\mathcal{B}$), and the other is called adversary($\mathcal{A}$). The IND game of public key cryptosystem is called the IND-CPA game under the chosen plaintext attack, which is defined as follows.

1) Initialization. A challenger generates the encryption system $\mathcal{E}$, and $\mathcal{A}$ gets the public key K_{Pub} of the system, which can be used to encrypt any plaintext;
2) Challenge. $\mathcal{A}$ chooses two same long plaintexts m_0 and m_1. The challenger randomly chooses $b \in \{0,1\}$ to encrypt $c^* = Enc_{K_{Pub}}(m_b)$, then send c^* to $\mathcal{A}$.
3) Guess. $\mathcal{A}$ outputs a guess $b' \in \{0,1\}$ of b. If $b = b'$, then output 1($\mathcal{A}$ wins the game); otherwise output 0.

Assume $Adv^{ind-cpa}_{\mathcal{A},\mathcal{E}}(\lambda)$ is the advantage of $\mathcal{A}$ winning $Game^{ind-cpa}_{\mathcal{A},\mathcal{E}}(\lambda)$. If there is a negligible function δ, such that

$$Adv^{ind-cpa}_{\mathcal{A},\mathcal{E}}(\lambda) = |Pr[Game^{ind-cpa}_{\mathcal{A},\mathcal{E}}(\lambda) = 1] - \frac{1}{2}| \leq \delta(\lambda),$$

then the scheme $\mathcal{E}$ is indistinguishably secure under the chosen plaintext attack, which means that the scheme is semantically secure.

3 Probabilistic RSA Encryption Algorithm

In this section, we modify the deterministic RSA encryption algorithm to a probabilistic RSA encryption algorithm by adding a random number into a ciphertext. The improved algorithm is still homomorphic, and it has higher security, which makes it more powerful in addressing many cryptography and information security problems.

3.1 Probabilistic RSA with Homomorphism

KeyGen.

1) Choose two large prime numbers p and q;
2) Compute $n = p \times q$ and $\varphi(n) = (p-1)(q-1)$, where $\varphi(n)$ is the value of Euler toient function of n;
3) Choose an integer e such that $1 < e < \varphi(n)$ and $\gcd(\varphi(n), e) = 1$;
4) Compute d such that $d \times e \equiv 1 \bmod \varphi(n)$;
5) The public key is (e, n), and the private key is (d, n).

Encrypt. To encrypt a message m, choose a random number $r(0 < r < n)$, and compute

$$c = (c_1, c_2) = (m^{r+e} \bmod n, m^{re} \bmod n).$$

Decrypt. To decrypt ciphertext c, compute

$$m = {c_1}^d \cdot {c_2}^{-d^2} \bmod n.$$

Generally, $r \in Z_n^*$, but r is selected by an encryption party, who does not know the factorization of n. Thus the encryption party can only select $r \in Z_n$. However, the analysis shows that the probability of $r \notin Z_n^*$ is negligible. Thus the encryption party just selects $r \in Z_n$. Moreover, d^2 can be processed before decryption, which will improve the efficiency of decryption.

3.2 Scheme Analyses

Correctness analysis. The encryption is to compute

$$c = (c_1, c_2) = (m^{r+e} \bmod n, m^{re} \bmod n).$$

Decryption is to compute

$$m = {c_1}^d \cdot {c_2}^{-d^2} \bmod n.$$

It is known by the decryption formula that obtaining the inverse of c_2 is the key of correctness proof. Probability of that the inverse of c_2 exists is not 100%, but the probability of that the inverse of c_2 does not exist is negligible. The reason is as follows [38].

Fact 1. Let $a \in \mathbb{Z}_n$. Then a is invertible if and only if $\gcd(a, n) = 1$.

Fact 2. (Euler's theorem)Let $n \geq 2$ be an integer. If $a \in \mathbb{Z}_n^*$, then $a^{\phi(n)} \equiv 1(\mathrm{mod} n)$.

This means that if the inverse of $(a \bmod n)$ exists, then $(a^{\phi(n)-1} \bmod n)$ is the inverse, because $a \cdot a^{\phi(n)-1} \bmod n = a^{\phi(n)} \bmod n = 1$.

In our scheme, no matter how to choose r and e, $c_2 \equiv m^{re} \bmod n \in Z_n$, the elements in Z_n are all invertible except multiples of p or multiples of q, because $\gcd(up, n) = p$ and $\gcd(vq, n) = q$, where u and v are integers. Then we will analyse the probability that c_2 has no inverse in the following. As mentioned above, up and vq have no inverse in Z_n. The number of up is $q-1$, and the number of vq is $p-1$. For example, when $n = p \times q = 5 \times 7 = 35$, the number of up is 6(5, 10, 15, 20, 25, 30), and the number of vq is 4(7, 14, 21, 28). These numbers(5, 10, 15, 20, 25, 30, 7, 14, 21, 28) have not inverse. Suppose that the bits of p is approximately equal to the bits of q, then the number of up and vq is approximately equal to $2(p-1)$. Thus the ratio of $2(p-1)$ to $n-1$ is the probability that c_2 has no inverse, that is,

$$\frac{2(p-1)}{n-1} \approx \frac{2}{q}.$$

Generally, because we need to ensure n is difficult to factorize, we must set the bits of p approximately equal to the bits of q and $(p \geq 2^{384}) \wedge (q \geq 2^{384})$. Thus the probability that c_2 has no inverse is less than $\frac{1}{2^{383}} = 2^{-383}$. Therefore the probability is negligible. Thus c_2 has inverse, and it can be obtained by computing $(c_2^{\phi(n)-1} \bmod n)$, or by following:

$$D(c) \equiv {c_1}^d \cdot {c_2}^{-d^2} \bmod n \equiv \frac{m^{(r+e)d} \bmod n}{m^{red^2} \bmod n} \equiv \frac{(m^{rd} \times m^{ed}) \bmod n}{(m^{rd})^{ed} \bmod n} \equiv \frac{m^{rd} \times m \bmod n}{(m^{rd}) \bmod n} = m \bmod n.$$

In conclusion, congruence expressions are not valid to division in the absence of inverses. However, c_2 has inverse in our scheme, so congruence expressions are valid to division, that is, both sides of the congruence expression are divisible. This kind of usage can also be seen in the Ref.[6], so it can be seen that congruence expressions are valid to division in the case of the inverse existing. This completes the proof of the correctness analysis.

Homomorphism analysis. The RSA probabilistic encryption algorithm keeps multiplicative homomorphism. The specific property is described as follows.

Evaluation. For given ciphertexts $E(M_1)$ and $E(M_2)$, compute

$$\begin{aligned} v &= E(M_1) \times E(M_2) \bmod n \\ &= (M_1^{r_1+e}, M_1^{r_1 e}) \times (M_2^{r_2+e}, M_2^{r_2 e}) \bmod n \\ &= (M_1^{r_1} M_2^{r_2} (M_1 M_2)^e \bmod n, (M_1^{r_1} M_2^{r_2})^e \bmod n) \\ &= (c_1, c_2). \end{aligned}$$

Decrypting v can obtain:

$$\begin{aligned} D(v) &= {c_1}^d \cdot {c_2}^{-d^2} \bmod n \\ &= \frac{[M_1^{r_1} M_2^{r_2} (M_1 M_2)^e]^d \bmod n}{[(M_1^{r_1} M_2^{r_2})^e]^{d^2} \bmod n} \\ &= \frac{M_1^{r_1 d} M_2^{r_2 d} (M_1 M_2)^{ed} \bmod n}{(M_1^{r_1 d} M_2^{r_2 d})^{ed} \bmod n} \\ &= \frac{M_1^{r_1 d} M_2^{r_2 d} M_1 M_2 \bmod n}{M_1^{r_1 d} M_2^{r_2 d} \bmod n} \\ &= M_1 M_2 \bmod n. \end{aligned}$$

Therefore the RSA variant is multiplicatively homomorphic, that is,

$$E(M_1) \times E(M_2) \bmod n \equiv E(M_1 \times M_2) \bmod n.$$

Security analysis. About the security of this scheme, we have the following theorem [39, 40].

Theorem 1. *If the RSA problem is difficult, then $\mathcal{E}$ has IND-CPA security, that is, the scheme is semantically secure. Assume $\mathcal{E}(Gen, Enc, Dec)$ is RSA variant. $\mathcal{A}$ is a polynomial time algorithm that attacks $\mathcal{E}$, and the advantage of $\mathcal{A}$ winning IND-CPA game is ξ. We can construct an algorithm $\mathcal{B}$ that can use $\mathcal{A}$ to solve the RSA problem.*

Proof. The challenger($\mathcal{B}$) of the RSA problem works as follows.

1) Inputs λ. Runs GenRSA(λ) and obtains (n, e, d). The public key is (n, e), and the private key is (n, d);
2) Sends system parameter λ and the public key (n, e) to $\mathcal{A}$;
3) Obtains M_0 and M_1 of $\mathcal{A}$;
4) Randomly selects $b \in \{0, 1\}$;
5) Assumes $C^* = (T_1 M^{e-1} \bmod n, T_2^e \bmod n)$ and sends C^* to $\mathcal{A}$;
6) Supposes that $b' \in \{0, 1\}$ is the guess of $\mathcal{A}$;
7) Outputs s'(If $b = b'$, then $s' = 0$; if $b \neq b'$, then $s' = 1$).

□

The probability of $\mathcal{B}$ winning RSA security game can be solved by Bayes formula as follows:

$$\begin{aligned} &Pr[s = s'] \\ &= Pr[s = 0]Pr[s = s'|s = 0] + Pr[s = 1]Pr[s = s'|s = 1] \\ &= \frac{1}{2}Pr[s' = 0|s = 0] + \frac{1}{2}Pr[s' = 1|s = 1] \\ &= \frac{1}{2}Pr[b = b'|s = 0] + \frac{1}{2}Pr[b \neq b'|s = 1]. \end{aligned} \quad (1)$$

When $s' = 0$, $\mathcal{B}$ sets $T = (T_1, T_2) = (M^{r+1} \bmod n, M^r \bmod n)$. At this point, the view of $\mathcal{B}$ submitted to $\mathcal{A}$ is indistinguishable from the view of $\mathcal{A}$ attacking $\mathcal{E}$ in the IND-CPA game. Therefore when $s' = 0$, the probability of $b = b'$ is equal to the probability of $\mathcal{A}$ winning the IND-CPA game, that is,

$$Pr[b = b'|s = 0] = \frac{1}{2} + \xi. \tag{2}$$

When $s' = 1$, $\mathcal{B}$ sets $T = \mathcal{R}_w = (\mathcal{R}_1, \mathcal{R}_2)$. Because R_w is uniformly distributed over Z_n, we can obtain that $(R_1 M^{e-1} \bmod n, R_2^e \bmod n)$ is uniformly distributed over (Z_n^*, Z_n^*), which is independent of n, M_0, M_1 and b. $(R_1 M^{e-1} \bmod n)$ and $(R_2^e \bmod n)$ are independent of M_0, M_1 and b. Therefore K_{Pub} and ciphertext C^* do not reveal any information about b, and guess b' outputed by $\mathcal{A}$ must be independent of b. Because the probability of $b = 0$ and $b = 1$ are both $1/2$, we can obtain

$$Pr[b \neq b'|s = 1] = \frac{1}{2}. \tag{3}$$

By Equations (1), (2) and (3), we can obtain

$$\begin{aligned} Pr[s = s'] &= \frac{1}{2}(\frac{1}{2} + \xi) + \frac{1}{2} \times \frac{1}{2} \\ &= \frac{1}{2} + \frac{1}{2}\xi. \end{aligned}$$

Therefore the advantage of $\mathcal{B}$ winning the game is

$$|Pr[s = s'] - \frac{1}{2}| = (\frac{1}{2} + \frac{1}{2}\xi) - \frac{1}{2} = \frac{\xi}{2}.$$

We are aware of that $\mathcal{B}$ can only win the game with negligible advantage, so $\xi/2$ is negligible, which implies ξ is also negligible. Therefore $\mathcal{A}$ can only win the IND-CPA game with the negligible advantage ξ.

Thus, using this scheme to encrypt any two same long plaintexts M_0 and M_1, the correspongding ciphertexts C_0 and C_1 are indistinguishable, that is, $C_0 \stackrel{c}{\equiv} C_1$.

4 Efficient Probabilistic RSA with Homomorphism

In Section 3, we modify the deterministic RSA encryption algorithm to a probabilistic RSA encryption algorithm, and maintain the homomorphism. However, encryption efficiency is reduced, so we introduce another variant. The new variant not only keeps homomorphism and semantic security but also greatly improves the efficiency of encryption.

4.1 Efficient Probabilistic RSA with Homomorphism

KeyGen.

1) Choose two large prime numbers p and q;
2) Compute $n = p \times q$ and $\varphi(n) = (p-1)(q-1)$, where $\varphi(n)$ is the value of Euler toient function of n;
3) Choose an integer e such that $1 < e < \varphi(n)$ and $\gcd(\varphi(n), e) = 1$;
4) Compute d such that $d \times e \equiv 1 \bmod \varphi(n)$;
5) The public key is (e, n), and the private key is (d, n).

Encrypt. To encrypt a message m, choose a random number $r (0 < r < n)$, and compute

$$c = (c_1, c_2) = (r^e \bmod n, rm^e \bmod n).$$

Decrypt. To decrypt ciphertext c, compute

$$M = (c_2 {c_1}^{-d})^d \bmod n.$$

4.2 Scheme Analyses

Correctness analysis. The encryption is to compute

$$E(M) = (c_1, c_2) = (r^e \bmod n, rm^e \bmod n).$$

Decryption is to compute

$${c_1}^d \bmod n \equiv (r)^{ed} \bmod n \equiv r \bmod n$$

and

$$(\tfrac{c_2}{r})^d \bmod n \equiv m \bmod n.$$

Homomorphism analysis. The efficient RSA probabilistic encryption algorithm keeps multiplicative homomorphism. The specific property is described as follows.

Evaluation. For given ciphertexts $E(M_1)$ and $E(M_2)$, compute

$$\begin{aligned} &E(M_1) \times E(M_2) \bmod n \\ &\equiv ({r_1}^e, r_1 {M_1}^e) \times ({r_2}^e, r_2 {M_2}^e) \bmod n \\ &\equiv ((r_1 r_2)^e, r_1 r_2 (M_1 M_2)^e) \bmod n \\ &\equiv E(M_1 \times M_2) \bmod n. \end{aligned}$$

Therefore the RSA variant is multiplicatively homomorphic.

Security analysis. The proof method [41] is similar to the proof of Theorem 1, and we omit it here.

5 Applications

5.1 Probabilistic Digital Signature Scheme

The rapid development of Internet, Internet of things and car networking animate booming development of e-commerce of the world. Using handwritten signatures will greatly reduce transaction efficiency of e-commerce, so more and more people want to replace handwritten signatures with fast and convenient digital signature for signing the agreement in real life to improve the trade efficiency Moreover, digital signature not only ensure the security and accuracy of data transmission, but also confirm the identity of both parties. Therefore the applications of digital signature are increasingly wide [42, 43], and the research of digital signature is meaningful. In this paper, we improve the security of the signature by introducing a random number, which is sufficient to resist the homomorphic attack. The specific signature scheme is as follows.

Sign: 1) Suppose that the message is m;
2) Generate signature $S_1 \equiv m^{r+d} \bmod n$;
3) Generate signature $S_2 \equiv m^{rd} \bmod n$;
4) Output (m, S_1, S_2).

Verify: 1) Obtain (m, S_1, S_2);
2) Compute $h_1 = {S_1}^e \bmod n$;
3) Compute $h_2 = {S_2}^{e^2} \bmod n$;
4) Compute $h' = (h_1, h_2) = h_1 {h_2}^{-1}$;
5) Compare whether $m = h'$. If $m = h'$, then accept the signature; otherwise reject the signature.

Correctness analysis. The signature process is:

$$s = (s_1, s_2) = (m^{r+d} \bmod n, m^{rd} \bmod n).$$

Verifying s can obtain:

$$\frac{{s_1}^e \bmod n}{{s_2}^{e^2} \bmod n} \equiv \frac{m^{(r+d)e} \bmod n}{m^{rde^2} \bmod n} \equiv \frac{(m^{re} \times m^{de}) \bmod n}{(m^{re})^{de} \bmod n} \equiv \frac{m^{re} \times m \bmod n}{m^{re} \bmod n} = m \bmod n.$$

5.2 Homomorphism Attack

Homomorphic attack refers to that a malicious attacker uses homomorphism to forge a new signature in order to achieve attacks. The deterministic RSA signature algorithm cannot resist homomorphism attack. It mainly has the following two attacks:

1) If the attacker knows the messages M_1 and M_2, and the corresponding signatures S_1 and S_2, then the attacker can forge signature $S = (M_1 \times M_2)^d \bmod n$ of message $M = (M_1 \times M_2) \bmod n$, because $S = (S_1 \times S_2) \bmod n = (M_1^d \times M_2^d) \bmod n = (M_1 \times M_2)^d \bmod n$;
2) If the attacker knows the messages M_1 and M_2, and the corresponding signatures S_1 and S_2, then the attacker can forge signature $S = (M_1^a \times M_2^b)^d \bmod n$ of message $M = (M_1^a \times M_2^b) \bmod n$, where a and b are positive integers, because $S = S_1^a S_2^b \bmod n = (M_1^d)^a \times (M_2^d)^b \bmod n = (M_1^a M_2^b)^d \bmod n$.

If M is a valuable piece of information, then the signature of M will be very important, and it is very dangerous for a malicious attacker to hold such an important signature. In order to resist the above two attacks, this paper proposes a new probabilistic signature scheme based on the first RSA variant. Because it is the application part, this paper just does an intuitive analysis here. Suppose that the attacker knows the two messages M_1 and M_2 and the corresponding signatures S and S'. If an attacker can forge the signatures $S^* = (S_1^*, S_2^*) = ((M_1 M_2)^{r+d}, (M_1 M_2)^{rd})$ of $M_1 M_2$ by multiplicatively transforming $S = (S_1, S_2) = (M_1^{r_1+d}, M_1^{r_1 d})$ and $S' = (S'_1, S'_2) = (M_2^{r_2+d}, M_2^{r_2 d})$, then the scheme cannot resist homomorphism attack. However, we can only obtain $S \times S' \bmod n = ((M_1 M_2)^d M_1^{r_1} M_2^{r_2}, (M_1^{r_1} M_2^{r_2})^d) \bmod n$ by multiplicative transformation, that is, the forged signatures cannot be verified. Therefore our signature algorithm can resist homomorphism attack.

With the RSA-blinding, a message provider can add a blind factor to the message, ask the signer to blind sign the message, and then remove the blind factor. This approach is not applicable for secure communication, because the sender can add a blind factor but the receiver cannot remove the blind factor unless the sender send the blind factor by a different channel. Therefore, RSA-blinding is mainly used to obtain non-determinstic blind signatures [1]. It cannot be used to secure a communication, nor can it be used in general protocols such as secure multiparty computations. Our RSA variants can be used to sign a message, to secure a communication, or to construct secure multiparty computation protocols. Using RSA-blinding signature, a malicious attacker may lure a signer to sign a message that hurts his benefit. Our probabilistic RSA can prevent this attack because our scheme can resist homomorphism attack.

To sum up, although the approach is not new, our constructions are completely new and have significant advantages. our probabilistic schemes with homomorphism can be used either to encrypt a message (to secure a communication) or to sign a message, or to construct general cryptographic protocols such as secure multiparty computation protocols. RSA blinding can only be used to make a blind signature. These are the advantages of our scheme.

5.3 Digital Commitment Scheme

Digital commitment is an important module of cryptography. Besides it can be widely used in constructing zero knowledge proof protocols and coin-tossing protocols, it also has important applications in real life, for example, confidential bidding. In addition, digital commitment

can be applied to electronic voting, electronic lottery and other aspects. Therefore studying more efficient digital commitment is of great significance. Generally, digital commitment scheme [44-46] is divided into the following two categories: the first is bit commitment, which means that the commitment information is limited to 0 and 1; the second is digital commitment, which means that the commitment information can be numbers or strings. In short, a digital commitment scheme is a two phase agreement with two parties taken part in. The two parties are the commitment maker and the receiver, and the two phases are commitment phase and revealing phase. The commitment maker achieves that the secret information is binded to a number through this protocol. The binding satisfies confidentiality and certainty.

However, the deterministic RSA encryption algorithm cannot be used to construct a digital commitment scheme, while the probabilistic RSA encryption algorithm can be used to construct the commitment scheme. Based on the second RSA variant, this paper proposes a non-malleable commitment scheme based on large prime factorization problem. The specific commitment is as follows:

KeyGen. Choose two private large primes p and q, and compute $n = p \times q$ and $\varphi(n) = (p-1)(q-1)$, where $\varphi(n)$ is the Euler toient function value of n. Two parties choose an integer e, where $1 < e < \varphi(n)$ and $\gcd(\varphi(n), e) = 1$.

Protocol 5.3.1 Non-malleable commitment scheme based on factorization.

Commitment phase. The commitment maker uniformly selects a random number of $r(0 < r < n)$, and compute

$$c(v) = (c_1, c_2) = (r^e \bmod n, rv^e \bmod n).$$

Revealing phase. The commitment maker sends r and v to the receiver, and the receiver verifies whether the following equation is true.

$$(r^e \bmod n, rv^e \bmod n) = (c_1, c_2).$$

If it is true, then accept the commitment; otherwise reject the commitment.

Confidentiality analysis. In this protocol, the commitment of any number is computationally indistinguishable from the commitment of other numbers. Assume commitments of v and $v+b$ respectively are

$$\begin{aligned} c(v) &= (c_1, c_2) \\ &= (r^e, rv^e \bmod n) \\ c(v+b) &= (c_1, c_2) \\ &= ((r+x)^e, (r+x)(v+b)^e \bmod n). \end{aligned}$$

The commitment maker uniformly selects a random number in commitment, so the commitment maker possibly selects r or $r+x$. However, the receiver cannot determine that which is the commitment of v, and which is a commitment of $v+b$. Because the above commitment values have been randomized, the receiver does not distinguish commitments between v and $v+b$.

Determinacy analysis. Because $c_1 = r^e \bmod n$, we can obtain that $r^e \bmod n$ is deterministic when the r is deterministic, that is, there is one-to-one correspondence between c_1 and r, which means that an attacker cannot forge r' such that $r'^e \bmod n = r^e \bmod n$. Analogously, because $\frac{rv^e}{r} \bmod n = v^e \bmod n$, we know that v^e is also deterministic. Thus the commitment scheme satisfies the requirement of certainty at the meaning of the computational feasibility.

Non-malleability analysis. The commitment information is $c(v)$. If the attacker wants to make a commitment of $v+b$ according to $c(v)$, he/she must know the value of the v. If the attacker wants to know the value of the v, then he/she need to factorize large number. However the problem of factoring large numbers is difficult, so this scheme is non-malleable.

6 Performance Analyses

6.1 Computational Complexity

In public key encryption system, the Paillier and the ElGamal are probabilistic encryption algorithms. Among them, Paillier's encryption algorithm and our two schemes are based on the same difficult problem, namely the factorization of large integers. ElGamal's encryption algorithm and our two schemes have same homomorphism, that is, multiplicative homomorphism. Because the schemes of our paper are mainly based on modulus exponentiations, we can measure the computation overhead of the algorithms by comparing modulus exponentiations. Suppose that the computation overhead of modulus n is x, the computation overhead of modulus n^2 is y, the computation overhead of modulus p is z, the computation overhead of modulus m is h, the computation overhead of modulus m^2 is k, and the computation overhead of modulus m^2-1 is t. The analysis of each scheme is shown in Table 1. To simplify the description, we define the RSA variant of Section 3 is PRSA 1, and the RSA variant of Section 4 is PRSA 2.

6.2 Experiments

In this section, we present two experimental results in terms of two RSA variants efficiency. The experimental settings are as follows, the operating system is Windows 10, CPU is Inter Core i5-6600 3.30GHz, and RAM is 8GB. We implement the schemes of this paper by using Java language and use the Experiment 1 and Experiment 2 to test the cost of PRSA 1 and PRSA 2. Execution

Table 1: Comparison of all solutions

	Scheme [33]	**PRSA 1**	**PRSA 2**	**Paillier's**	**ElGamal's**
Computation	$2x + h + k + t$	$4x$	$3x$	$3y + x$	$3z$
Number of keys	7	3	3	3	2

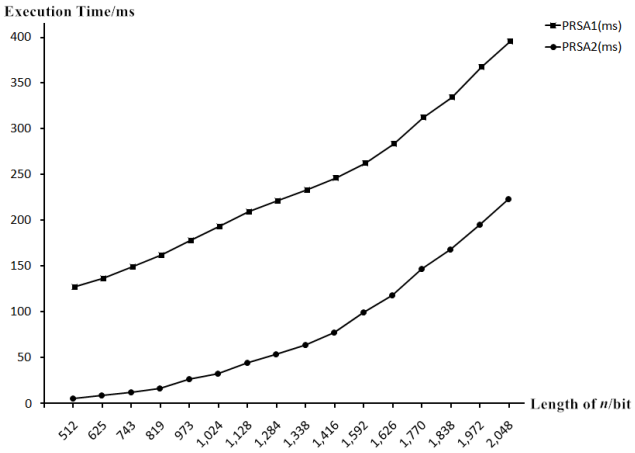


Figure 1: Comparison of the implementation of our two RSA variants($m = 20$)

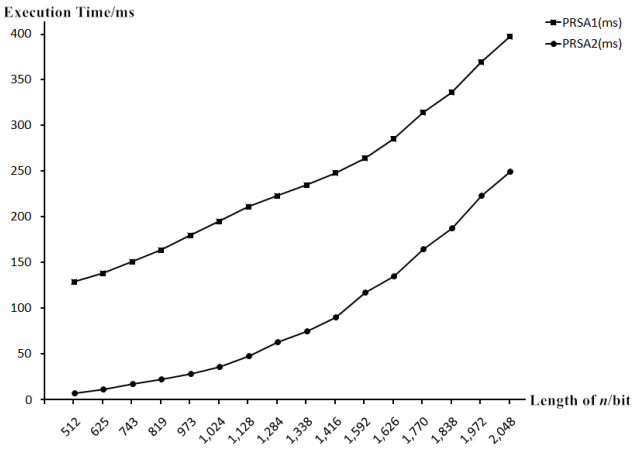


Figure 2: Comparison of the implementation of our two RSA variants($m = 2000$)

time of the four protocols verifies the computational complexity(Our implementation is not to compare the performance of our scheme with that of other schemes, but to show that our scheme is practical, so we did not use standard benchmark implementation [26, 45, 46, 51]).

Experiment 1. This experiment adopts control variable method. The length of n is independent variable. The range of n is [512, 2048]. The confidential datas are 20 and 2000. We implement PRSA1 and PRSA2, and each execution time in the experiment is the average time of 100 times encrypting and decrypting same plaintext. The results are shown in Figure 1 and Figure 2.

Figure 1 and Figure 2 show that the execution time grows as the length of the modulus increase, which has nothing to do with the length of confidential data. This is because that the execution time of the other operations is negligible compared with the execution time of the modulus exponentiations. In addition, the

modulus exponentiation operation is also uncertain. After modulus exponentiation operation, a small number may become a large number, and a large number may become a small number. Thus the length of confidential data does not affect the efficiency of the algorithm. The results show that the execution time of PRSA 1 is much larger than that of PRSA 2 under the condition of the limited length of confidential data, because modulus exponentiations of PRSA 1 is more than that of PRSA 2.

Experiment 2. This experiment adopts control variable method. The length of n is independent variable. The range of n is [512, 2048]. The confidential datas are 20 and 2000. We implement the Paillier, PRSA1, the ElGamal and PRSA2, and each execution time in the experiment is the average time of 100 times encrypting and decrypting same plaintext. The results are shown in Table 2.

The experimental datas show that the execution time of Paillier algorithm is much larger than that of PRSA 1 and PRSA 2, because the calculation of modulus n^2 of Paillier is more than the calculation of modulus n of PRSA 1 and PRSA 2. It can be seen that PRSA 2 is the most efficient.

7 Conclusion

RSA algorithm is of practical significances in information security, and it also has wide applications in public key cryptography. RSA algorithm can serve as the basic module of many cryptographic protocols, and it can be even widely used to guarantee secrecy communications, and confidentiality of the Internet, the Internet of things, and car networking business. In order to improve the security of RSA encryption algorithm and extend its applications, this paper modifies it to probabilistic encryption algorithms with semantic security. The RSA variants can resist homomorphic attack, which can be used for probabilistic encryptions, probabilistic signatures and digital

Table 2: Comparison of all solutions

Performance	Paillier's	PRSA 1	ElGamal's	PRSA 2
Alice(ms)	103	87	21	18
Number of keysBob(ms)	138	129	11	9
Total(ms)	241	216	32	27

commitments. Moreover, the variants can be widely used in designing various security protocols and it also provides a new and effective tool for designing cryptographic protocols. Theoretical analyses and experiments show that the algorithms are secure and efficient.

Acknowledgments

The authors would like to thank the anonymous reviewers for detailed and valuable comments. This work is supported by the National Natural Science Foundation of China (Grant no. 61272435).

References

[1] Bellare, Namprempre, Pointcheval, *et al.* "The one-more-RSA-inversion problems and the security of Chaum's blind signature scheme," *Journal of Cryptology*, vol. 16, no. 3, pp. 185-215, 2003.

[2] M. Bellare and P. Rogaway "Optimal asymmetric encryption," *Lecture Notes in Computer Science*, vol. 950, no. 6, pp. 92-111, 1994.

[3] D. Boneh and M. Franklin, "Identity-based encryption from the weil pairing," in *Advances in Cryptology (CRYPTO'01)*, pp. 213-229, 2001.

[4] Z. J. Cao and M. L. Liu, "Improvement of signature scheme based on strong RSA," *Chinese Journal of Computers*, vol. 29, no. 9, pp. 1617-1621, 2006.

[5] C. C. Chang, M. S. Hwang, "Parallel computation of the generating keys for RSA cryptosystems", *Electronics Letters*, vo1. 32, no. 15, pp. 1365–1366, 1996.

[6] G. L. Chen, "Mathematical foundations in information security," *Beijing: Tsinghua University Press*, 2004.

[7] M. M. V. Deshmukh, P. A. Tijare and S. N. Sawalkar, "A survey on privacy preserving data mining techniques for clinical decision support system," *International Research Journal of Engineering and Technology*, vol. 3, no. 5, pp. 2064-2069, 2016.

[8] R. S. Dhakar, A. K. Gupta and P. Sharma, "Modified RSA encryption algorithm (MREA)," in *Second International Conference on Advanced Computing and Communication Technologies*, pp. 426-429, 2012.

[9] W. Diffie and M. E. Hellman, "New directions in cryptography," *IEEE Transactions on Information Theory*, vol. 22, no. 6, pp. 644-654, 1976.

[10] T. Elgamal, "A public key cryptosystem and a signature scheme based on discrete logarithms," *Journal of IEEE Transactions on Information Theory*, vol. 31, no. 4, pp. 469-472, 1984.

[11] K. El-Makkaoui, A. Ezzati and A. Beni-Hssane, "Cloud-RSA: An enhanced homomorphic encryption scheme," in *Europe and MENA Cooperation Advances in Information and Communication Technologies*, pp. 471-480, 2017.

[12] M. Fischlin, D. Schroder "Security of blind signatures under aborts," in *International Conference on Practice and Theory in Public Key Cryptography*, pp. 297-316, 2009.

[13] M. Fischlin and R. Fischlin, "Efficient non-malleable commitment schemes," in *Annual International Cryptology Conference*, pp. 413-431, 2000.

[14] E. Fujisaki, T. Okamoto, D. Pointcheval, *et al.*, "RSA-OAEP is secure under the RSA assumption," *Journal of Cryptology*, vol. 17, no. 2, pp. 81-104, 2004.

[15] C. Gentry, "A fully homomorphic encryption scheme," *Stanford University*, 2009. ISBN: 978-1-109-44450-6

[16] S. Goldwasser "Lecture notes on cryptography," *Cite Seer X*, 1996. (`http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.56.4314&rep=rep1&type=pdf`)

[17] O. Goldreich, "Secure multi-party computation," *Journal of Manuscript Preliminary Version*, 1998. (`https://www.researchgate.net/profile/Oded_Goldreich/publication/2934115_Secure_Multi-Party_Computation/links/00b7d52bb04f7027d4000000.pdf`)

[18] S. Goldwasser and S. Micali, "Probabilistic encryption and how to play mental poker keeping secret all partial information," in *Proceeding of the 14th ACM Symposium on the Theory of Computer*, pp. 365-377, 1982.

[19] S. Goldwasser and S. Micali, "Probabilistic encryption," *Journal of Computer and System Science*, vol. 28, no. 1, pp. 270-299, 1994.

[20] S. Goldwasser, S. Micali and C. Rackoff, "The knowledge complexity of interactive proof systems," *SIAM Journal on Computing*, vol. 18, no. 1, pp. 186-208, 1989.

[21] L. M. Gong, S. D. Li, D. S. Wang and J. W. Dou, "A randomized coding of plaintext encryption scheme," *Journal of Software*, vol. 2017, no. 2, pp. 372-383, 2017.

[22] L. M. Gong, S. D. Li and J. W. Dou, "A public-key cryptosystem secure against adaptive chosen ciphertext attack," *Journal of Cryptologic Research*, vol. 3, no. 1, pp. 42-55, 2016.

[23] L. M. Gong, S. D. Li, Q. Mao, *et al.*, "A homomorphic encryption scheme with adaptive chosen ciphertext security but without random oracle," *Theoretical Computer Science*, vol. 609, no. 1, pp. 253-261, 2016.

[24] J. Hoffstein, J. Pipher and J. H. Silverman, "NTRU: A ring-based public key cryptosystem," *International Algorithmic Number Theory Symposium*, pp. 267-288, 1998.

[25] N. Koblitz, "Elliptic curve cryptosystems," *Mathematics of Computation*, vol. 48, no. 177, pp. 203-209, 1987.

[26] P. C. Kocher, "Timing attacks on implementations of diffie-hellman, RSA, DSS, and other systems," in *International Cryptology Conference on Advances in Cryptology*, pp. 104-113, 1996.

[27] Y. F. Li, Q. Liu, T. Li, *et al.*, "An improved algorithm for batch RSA," *Journal of Chinese Mini-Micro Computer Systems*, vol. 33, no. 1, pp. 64-70, 2012.

[28] Q. Li and J. Y. Zhang, "An improved fast RSA algorithm," *Journal of Chinese Mini-Micro Computer Systems*, vol. 22, no. 1, pp. 70-72, 2001.

[29] S. D. Li and D. S. Wang, "Efficient secure multiparty computation based on homomorphic encryption," *Acta Electronica Sinica*, vol. 41, no. 4, pp. 798-803, 2013.

[30] S. D. Li, S. F. Zhou, Y. M. Guo, *et al.*, "Secure set computing in cloud environment," *Journal of Software*, vol. 27, no. 6, pp. 1549-1565, 2016.

[31] S. D. Li, J. W. Dou and D. S. Wang, "Survey on homomorphic encryption and its applications to cloud security," *Journal of Computer Research and Development*, vol. 52, no. 6, pp. 1378-1388, 2015.

[32] S. D. Li and D. S. Wang, "Modern cryptography: Theory, method and research forefront," *Science Press*, 2009.

[33] G. Liu and T. F. Jiang, "Research on homomorphic encryption technology and the applications of it in IOT," *Journal of Netinfo Security*, vol. 2011, no. 5, pp. 61-64, 2011.

[34] Y. H. Liu, Z. K. Dai and H. Li, "An improved public-key algorithm based on RSA," *Journal of Sichuan University (Natural Science Edition)*, vol. 42, no. 4, pp. 760-764, 2005.

[35] A. J. Menezes, P. C. V. Oorschot, S. A. Vanstone, "Handbook of applied cryptography," *CRC Press*, pp. 816, 1996. ISBN: 0-8493-8523-7

[36] V. S. Miller, "Use of elliptic curves in cryptography," in *Conference on the Theory and Application of Cryptographic Techniques*, pp. 417-426, 1985.

[37] M. Naor, "Bit commitment using pseudorandomness," *Journal of Cryptology*, vol. 4, no. 2, pp. 151-158, 1991.

[38] T. Okamoto and S. Uchiyama, "A new public-key cryptosystem as secure as factoring," in *International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 308-318, 1998.

[39] P. Paillier, "Public-key cryptosystems based on composite degree residuosity classes," in *International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 223-238, 1999.

[40] T. P. Pedersen, "Non-interactive and information-theoretic secure verifiable secret sharing," in *Annual International Cryptology Conference, Springer Berlin Heidelberg*, pp. 129-140, 1991.

[41] M. Qiu, S. S. Luo, W. Liu, *et al.*, "A solution of secure multi-party multi-data ranking problem based on RSA encryption scheme," *Acta Electronica Sinica*, vol. 37, no. 5, pp. 1119-1123, 2009.

[42] M. O. Rabin, "How to exchange secrets with oblivious transfer," *IACR Cryptology Eprint Archive*, 2005. (`https://www.semanticscholar.org/paper/How-To-Exchange-Secrets-with-Oblivious-Transfer-Rabin/1d2a3436fc7ff4b964fa61c0789df19e32ddf0ed`)

[43] M. O. Rabin, "Digitalized signatures and public-key functions as intractable as factorization," *Journal of Massachusetts Institute of Technology*, 1979. (`https://dl.acm.org/citation.cfm?id=889813`)

[44] R. L. Rivest, A. Shamir and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," *Journal of Communications of the ACM*, vol. 26, no. 1, pp. 96-99, 1978.

[45] M. A. Sadikin, R. W. Wardhani, "Implementation of RSA 2048-bit and AES 256-bit with digital signature for secure electronic health record application," in *International Seminar on Intelligent Technology and ITS Applications*, pp. 387-392, 2017.

[46] H. Siregar, E. Junaeti, T. Hayatno, "Implementation of digital signature using Aes and Rsa algorithms as a security in disposition system af letter," in *IOP Conference Series: Materials Science and Engineering*, vol. 180, no. 1, pp. 012-055, 2017.

[47] A. Shamir, "How to share a secret," *Communications of the ACM*, vol. 22, no. 11, pp. 612-613, 1979.

[48] V. Shoup, "OAEP reconsidered (extended abstract)," *Proceeding of Cryptography*, vol. 15, no. 4, pp. 239-259, 2001.

[49] K. Suveetha and T. Manju, "Ensuring confidentiality of cloud data using homomorphic encryption," *Indian Journal of Science and Technology*, vol. 9, no. 8, pp. 1-7, 2016.

[50] L. C. Wang, J. Li and H. Ahmad, "Challenges of fully homomorphic encryptions for the internet of things," *Journal of Ieice Transactions on Information and Systems*, vol. 99, no. 8, pp. 1982-1990, 2016.

[51] H. Xia, Q. Pei, Y. Xi. "The analysis and research of freak attack based on OpenSSL," in *International Conference on Information Engineering for Mechanics and Materials*, 2016.

[52] M. S. Yu and H. Zou, "An improved RSA public key cryptosystem," *Journal of Dalian University of Technology*, vol. 43, no. 1, pp. 50-52, 2003.
[53] B. Yang, "Modern cryptography," *Beijing: Tsinghua University Press*, 2007.
[54] Y. B. Zhou, Z. F. Zhang, S. H. Qin, *et al.*, "A fair exchange protocol based on RSA signature scheme," *Journal of Software*, vol. 15, no. 7, pp. 1049-1055, 2004.

Biography

Yaling Geng was born in 1993. She is currently pursuing the M.S. degree with School of Computer Science in Shaanxi Normal University. Her research interests focus on modern cryptography and information security.

Shundong Li was born in 1963. He received the Ph.D. degree in Department of computer science and technology from Xian JiaoTong University in 2003. He is now a Professor with School of Computer Science in Shaanxi Normal University. His research interests focus on modern cryptography and secure multi-party computation.

Sufang Zhou was born in 1990. She is currently pursuing the PH.D. degree with School of Computer Science in Shaanxi Normal University. Her research interests focus on secure multi-party computation and information security.

Cryptanalysis of An Improved Predicate Encryption Scheme from LWE

Chengbo Xu
School of Mathematical Sciences, University of Jinan
No. 336, Nanxinzhuang West Road, Jinan 250022, Shandong, P. R. China
(Email: cbqysy@163.com)

(Received Apr. 1, 2018; Revised and Accepted Aug. 21, 2018; First Online June 5, 2019)

Abstract

Predicate encryption scheme is a paradigm which provides fine-grained access control and has attractive applications. In 2017, Brakerski, Tsabary, Vaikuntanathan, and Wee (TCC 2017) proposed a new LWE based predicate encryption scheme in order to overcome drawbacks in the scheme proposed by Gorbunov, Vaikuntanathan and Wee (CRYPTO 2015). In this paper, We analyze this scheme and provide two practical attacks to show that the scheme (TCC 2017) is insecure under the full attribute hiding security model. These two attacks mainly exploit several homomorphic and linear properties in the construction. This illustrates that in order to construct full attribute hiding secure predicate encryption scheme these weak properties must be bypassed.

Keywords: Functional Encryption; Lattice with Error (LWE); Predicate Encryption

1 Introduction

With the emergence and development of cloud computing and other complex networks, considerable progress has been witnessed recently in the field of computing on encrypted data. A number of concepts and constructions of cryptographic primitives have turned out, such as Attribute Based Encryption [3,7,8,13,15,19,21,24,25], Fully Homomorphic Encryption [12, 14, 17, 18], Functional Encryption [1, 2, 4, 9, 16, 23].

Among them, the notion of fully homomorphic encryption permits arbitrary computation on encrypted data, but still restricts decryption to be *all or nothing* as traditional notions of public key encryption. However, *Functional encryption* [9], attribute based encryption [8, 19], provides a satisfying solutions to this problem in theory. Two features provided by functional encryption are fine-grained access and computing on encrypted data. The fine-grained access part is formalized as a cryptographic notion, named *predicate encryption* [10, 11, 20, 22]. In predicate encryption system, ciphertext ct is associated with descriptive attribute values a in addition to plaintexts μ while secret key sk_f is associated with a predicate f. A user holding the key sk_f can decrypt ciphertext ct if and only if $f(a) = 0$.

In the literature, The security requirement for predicate encryption scheme can be formalized in two ways. The basic one is the definition of weak attribute-hiding, which enforces privacy of a and the plaintext amidst multiple unauthorized secret key queries: an adversary holding secret keys for different query predicates learns nothing about the attribute x and the plaintext if none of them is individually authorized to decrypt the ciphertext. The second, called full attribute-hiding, requires that a remains hidden given an unbounded number of keys, which may comprise of both authorized and unauthorized keys.

Recently, Gorbunov, Vaikuntanathan and Wee [20] constructed a predicate encryption scheme for all circuits (of an a-priori bounded polynomial depth) from the LWE assumption. But the construction only achieved the weak attribute-hiding security. Two sources of leakage in the scheme prevent its construction from achieving the full attribute-hiding property. Later, Agrawal [2] indeed exploited the two sources of leakage to recover the attribute a under full attribute-hiding attacks. Based on these, Brakerski *etc.* [11] proposed an improved predicate encryption scheme by feat of the new "Dual Use" technique, that is, using the same LWE secret for the FHE [20] and the ABE [8]. In this paper, we cryptanalyze this improved scheme and show that it still does not achieve the full attribute-hiding security.

Our Contributions: We provide two polynomial time attacks to show that the Brakerski *etc.*'s predicate encryption scheme [11] is still not secure under the full attribute-hiding attacks.

1) Our first attack is inspired by the attack method in [2] which is designed to attack the inner product predicate encryption scheme [4] mainly using the inherent property of linearity in the inner product operation. However, the Brakerski *etc.*'s predicate encryption scheme we considered here, is designed for general predicates described by polynomial-size circuits, instead of only inner product predicate. Conse-

quently, two barriers prevent applying the attack into Brakerski *etc.*'s scheme directly. Fortunately, we find and prove two homomorphic properties which conquer above two barriers and make the attack practical.

2) Our second attack is based on the following three observations: The first one is that when running the ciphertexts homomorphic evolution algorithm in [8], the error growth is linear in the corresponding original errors. The second is that when running the GSW homomorphic evaluation algorithm, the error growth is also linear in the corresponding original errors. More importantly, the coefficients in these two linear combination are both public in view of the adversary. The last observation is that by construction of the scheme in [11], the adversary is able to obtain a set of linear equations over all the original errors given a 1-key. Thus, by requesting sufficient 1-keys, the attacker will solve this linear system to recover the errors used in encryption, which lead to recovery of the predicate a.

2 Preliminaries

Notation. Let λ be the security parameter, and let PPT denote probabilistic polynomial time. We use bold uppercase letters to denote matrices $\mathbf{M}$, and bold lowercase letters to denote vectors v. We write $[n]$ to denote the set $\{1, ..., n\}$, and $|t|$ to denote the number of bits in the string t. We denote the i-th bit s by $s[i]$. We say a function $negl(\cdot) : N \to (0, 1)$ is negligible, if for every constant $c \in N$, $negl(n) < n^{-c}$ for sufficiently large n.

2.1 Predicate Encryption

We recall the syntax and security definition of *predicate encryption* (PE) [4,22]. PE can be regarded as a generalization of attribute based encryption. A PE scheme PE with respect to an attribute universe A, predicate universe C and a message universe M consists of four algorithms $\Pi = (Setup, Keygen, Enc, Dec)$:

$Setup(1^\lambda, A, C, M)$: On input the security parameter λ, the setup algorithm outputs public parameters mpk and master secret key msk.

$keygen(msk, C)$: On input the master secret key msk and a predicate $C \in C$, the key generation algorithm outputs a secret key sk_C.

$Enc(mpk, a, \mu)$: On input the public parameter mpk and an attribute/message pair (a, μ), it outputs a ciphertext ct.

$Dec((sk_C, C), ct)$: On input the secret key sk_C and a ciphertext ct, it outputs the corresponding plaintext μ if $C(a) = 1$; otherwise, it outputs $\perp$.

Definition 1 (Correctness). *We say the PE scheme described above is correct, if for any* $(msk, mpk) \leftarrow Setup(1^\lambda)$, *any message* μ, *any predicate* $C \in C$, *and attribute* $a \in A$ *such that* $C(a) = 0$, *we have* $Dec(sk_C, ct) = \mu$, *where* $sk_C \leftarrow Keygen(msk, C)$ *and* $ct \leftarrow Enc(mpk, a, \mu)$.

Security. The model $Expt_A^{PE}(1^\lambda)$ for defining the fully attribute-hiding security of PE against adversary A (under chosen plaintext attacks) is given as follows:

1) *Setup* is run to generate keys mpk and msk, and mpk is given to A.
2) A may adaptively make a polynomial number of key queries for predicate functions, f. In response, A is given the corresponding key $sk_f \xleftarrow{R} Keygen(msk, f)$.
3) A outputs challenge attribute vector $(a^{(0)}, a^{(1)})$ and challenge plaintexts $(\mu^{(0)}, \mu^{(1)})$, subject to the following restrictions:
   - $f(a^{(0)}) \neq 0$ and $f(a^{(1)}) \neq 0$ for all the key queried predicate, f.
   - Two challenge plaintexts are equal, i.e., $\mu^{(0)} = \mu^{(1)}$, and any key query f satisfies $f(a^{(0)}) = f(a^{(1)})$, i.e., one of the following conditions.
     - $f(a^{(0)}) = 0$ and $f(a^{(1)}) = 0$;
     - $f(a^{(0)}) \neq 0$ and $f(a^{(1)}) \neq 0$,
4) A random bit b is chosen. A is given $ct_{a^{(b)}} \xleftarrow{R} Enc(mpk, \mu^{(b)}, a^{(b)})$.
5) The adversary may continue to issue a polynomial number of key queries for additional predicate, f, subject to the restriction given in Step 3. A is given the corresponding key $sk_f \xleftarrow{R} Keygen(mpk, msk, f)$.
6) A outputs a bit b', and wins if $b' = b$.

The advantage of adversary A in attacking a PE scheme PE is defined as:

$$Advantage_A(1^\lambda) = \left| \Pr[b^* = b'] - \frac{1}{2} \right|,$$

where the probability is over the randomness of the challenger and adversary.

Definition 2 (Fully attribute-hiding). *We say an PE scheme PE is fully attribute-hiding against chosen-plaintext attacks in adaptive attribute setting, if for all PPT adversaries A engaging in experiment* $Expt_A^{PE}(1^\lambda)$, *we have*

$$Advantage_A(1^\lambda) \leq negl(\lambda).$$

2.2 Gadget Matrix

We now recall the gadget matrix [5,23], and the extended gadget matrix technique appeared in [6], that are important to our construction.

Definition 3. *Let $m = n \cdot \lceil \log q \rceil$, and define the gadget matrix*

$$G_{n,2,m} = g \otimes I_n \in Z_q^{n \times m}$$

where vector $g = (1, 2, 4, ..., 2^{\lfloor \log q \rfloor}) \in Z_q^{\lceil \log q \rceil}$, and $\otimes$ denotes tenser product. We will also refer to this gadget matrix as "powers-of-two" matrix. We define the inverse function $G^{-1}_{n,2,m} : Z_q^{n \times m} \to \{0,1\}^{m \times m}$ which expands each entry $a \in Z_q$ of the input matrix into a column of size $\lceil \log q \rceil$ consisting of the bits of binary representations. We have the property that for any matrix $A \in Z_q^{n \times m}$, it holds that $G_{n,2,m} \cdot G^{-1}_{n,2,m}(A) = A$.

2.3 GSW Homomorphic Encryption Scheme

The GSW scheme [5,18] is parameterized by a dimension n, a modulus q with $l = \lceil \log_2 q \rceil$, and some error distribution χ over Z which we assume to be subGaussian. Formally, we describe the scheme as follows:

- GSW.Gen (choose $\overline{s} \leftarrow \chi^{n-1}$ and output secret key $s = (\overline{s}, 1) \in Z^n$).
- GSW.Enc $(s, \mu \in Z)$: choose $\overline{C} \leftarrow Z_q^{(n-1) \times nl}$ and $e \leftarrow \chi^m$, let $b^T = e^t - \overline{s}^T \overline{C} (\mathrm{mod} q)$, and output the cphertext

$$C = \begin{bmatrix} \overline{C} \\ b^T \end{bmatrix} + \mu G$$

  where G is the gadget matrix. Notice that $s^T C = e^T + \mu \cdot s^T G (\mathrm{mod} q)$.
- GSW.Dec(s, C): Let c be the penultimate column of C, and output $\mu = \lfloor \langle s, c \rangle \rceil_2$.
- GSW.Eval(C_1, C_2):
  - Homomorphic addition: $C_1 \boxplus C_2 = C_1 + C_2$.
  - Homomorphic multiplication: $C_1 \boxdot C_2 \leftarrow C_1 \cdot G^{-1}(C_2)$, and is right associative.

2.4 Lattice Evolution

The following lemma is an abstraction of the evaluation procedure that developed in a long sequence of works [3, 5, 8, 11, 18, 20]. Here we use the formalism as in [11].

Lemma 1. *There exist efficient deterministic algorithms EvalF and EvalFX such that for all $n, q, l \in N$, and for any sequence of matrices $(B_1, \cdots, B_l) \in (Z_q^{n \times n \lceil \log q \rceil})^l$, for any depth-$d$ Boolean circuit $f : \{0,1\}^l \to \{0,1\}$ and for every $x = (x_1, \cdots, x_l) \in \{0,1\}^l$, the following properties hold.*

- *The outputs $H_f = EvalF(f, B_1, \cdots, B_l)$ and $H_{f,x} = EvalFX(f, x, B_1, \cdots, B_l)$ are both matrices in $Z^{(ln\lceil \log q \rceil)} \times n \lceil \log q \rceil$;*
- *It holds that $\|H_f\|_\infty$, $\|H_{f,x}\|_\infty \leqslant (n \log q)^{O(d)}$;*
- *It holds that $[B_1 - x_1 G \| \cdots \| B_l - x_l G] \cdot H_{f,x} = [B_1 \| \cdots \| B_l] \cdot H_f - f(x) G (\mathrm{mod} q)$.*

Construction of algorithms EvalF and EvalFX:

☐ For an addition gate $f(x_1, \cdots, x_k) = x_1 + \cdots + x_k$,

$$EvalF(f, B_1, \cdots, B_k) = \begin{bmatrix} E & \cdots & E \end{bmatrix}^T$$
$$EvalFX(f, x, B_1, \cdots, B_k) = \begin{bmatrix} E & \cdots & E \end{bmatrix}^T$$

where E is the identity matrix.

☐ For a multiplication gate $f(x_1, \cdots, x_k) = x_1 x_2 \cdots x_k$,

$$EvalF(f, B_1, \cdots, B_k) = \begin{bmatrix} O \\ \vdots \\ O \\ G^{-1}(-B_{k-1} G^{-1}(\cdots G^{-1}(-B_2 G^{-1}(-B_1)))) \end{bmatrix}$$

$$EvalFX(f, x, B_1, \cdots, B_k) = \begin{bmatrix} x_2 x_3 \cdots x_k E \\ x_3 x_4 \cdots x_k G^{-1}(-B_1) \\ \vdots \\ G^{-1}(-B_{k-1} G^{-1}(\cdots G^{-1}(-B_2 G^{-1}(-B_1)))) \end{bmatrix}$$

where E is the identity matrix.

☐ For a general circuit f which has l input wires, we construct the required matrices inductively input to output gate-by-gate.

3 Review of the BTVW Predicate Encryption Scheme Using Dual-Use Technique

In this section, we provide a brief overview of the BTVW predicate encryption scheme using Dual-Use technique [11].

We write $\overline{G} \in Z_q^{n \times (n+1) \log q}$ to denote all but the last row of G which is the gadget matrices in $Z_q^{(n+1) \times (n+1) \log q}$. Given a circuit computing a function $f : \{0,1\}^l \to \{0,1\}$, and GSW FHE encryptions $\Psi := (\Psi_1, \cdots, \Psi_l)$ of $x_1, \cdots, x_l$,we write Ψ_f to denote fhe.eval(f, Ψ). Recalling syntax of GSW, Ψ_f is a matrix, and we denote the last row of Ψ_f as $\underline{\Psi}_f$, all but the last row of Ψ_f as $\overline{\Psi}_f$. In addition, we denote the circuit that computes $\Psi \mapsto \overline{\Psi}_f$ as $\hat{f}$, namely it takes as input the bits of Ψ and outputs the matrix $\overline{\Psi}_f$.

We let $e \xleftarrow{\sigma} Z^m$ denote the process of sampling a vector e where each of its entries is drawn independently from the discrete Gaussian with mean 0 and standard deviation σ over Z.

- $Setup(1^\lambda, 1^l, 1^d)$: sample (B, T_B) where $B \in Z_q^{n\times(n+1)\log q}$ and T_B denotes the trapdoor for B. Pick $B_j \xleftarrow{\$} Z_q^{n\times(n+1)\log q}$ and $p \xleftarrow{\$} Z_q^n$. Output

$$mpk := (B, \{B_j\}_{j\in[L]}, p), \qquad msk := (T_B)$$

where $L = l(n+1)^2 \log^2 q$.

- $Enc(mpk, x, M \in \{0,1\})$: pick $s \xleftarrow{\$} Z_q^n, e, e_0, e_j \xleftarrow{\sigma} Z^m, e' \xleftarrow{\$} Z, R_i \in \{0,1\}^{(n+1)\log q\times(n+1)\log q}$ and compute

$$\Psi_i := \begin{pmatrix} B \\ s^T B + e^T \end{pmatrix} R_i + x_i G.$$

Parse $\Psi := [\Psi_1|\cdots|\Psi_l]$ as its binary representation $\psi_1, \cdots, \psi_L$. Compute

$$c_{in}^T := s^T B + e_0^T, \qquad c_j^T := s^T[B_j - \psi_j \overline{G}] + e_j^T$$

and $c_{out} := s^T p + e' + M \cdot \lfloor q/2 \rfloor (\mod q)$. Set the PE ciphertext as follows:

$$ct := (\Psi, c_0, \{c_j\}_{j\in[L]}, c_{out}).$$

- $KeyGen(msk, f)$: Let $\hat{f}$ denote the circuit computing $\Psi \mapsto \overline{\Psi}_f$ and

$$H_{\hat{f}} := EvalF(\hat{f}, \{B_j\}_{j\in[L]}), B_{\hat{f}} := [B_1|\cdots|B_L]\cdots H_{\hat{f}}$$

Sample a short sk_f using T_B such that

$$[B|B_{\hat{f}}] \cdot sk_f = p.$$

Output sk_f.

- $Dec((sk_f, f), ct)$: Let $\hat{f}$ denote the circuit computing $\Psi \mapsto \overline{\Psi}_f$ and compute:

$$\Psi_f := \hat{f}(\Psi),$$

$$H_{\hat{f},\Psi} := EvalFX(\hat{f}, \Psi, \{B_j\}_{j\in[L]}),$$
$$c_{\hat{f}}^T := [c_1^T|\cdots|c_L^T] \cdot H_{\hat{f},\Psi} + \underline{\Psi}_f.$$

Output the MSB of $c_{out} - [c_{in}^T|c_{\hat{f}}^T] \cdot sk_f$.

4 Attack #I

In this section,we provide an attack to demonstrate that the predicate encryption scheme reviewed above is insecure against an adversary that requests 1-keys.

Case 1. Say the attacker requests keys for functions f_1 and f_2 such that for the challenge x it holds that:

$$f_1(x) = 0, \qquad f_2(x) = 0.$$

Then, by functionality, the attacker must learn two linear equations in the challenge x but must not learn anything more. Now, by the construction in [11], we can compute matrices B_{f_1} and B_{f_2} from the master public parameter mpk as follows:

$$B_{f_1} = EvalF(B_1, \cdots, B_L, \hat{f}_1),$$

$$B_{f_2} = EvalF(B_1, \cdots, B_L, \hat{f}_2),$$

where $\hat{f}_1$ and $\hat{f}_2$ denote circuits that compute $\Psi \mapsto \overline{\Psi}_{f_1}$ and $\Psi \mapsto \overline{\Psi}_{f_2}$ repectively. Then, we have the following equations:

$$[B|B_{f_1}] \begin{bmatrix} r_1 \\ r_2 \end{bmatrix} = p(\mod q),$$

$$[B|B_{f_2}] \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = p(\mod q).$$

Hence,

$$[B|B_{f_1}|B_{f_2}] \begin{bmatrix} r_1 - u_1 \\ r_2 \\ -u_2 \end{bmatrix} = 0(\mod q).$$

Thus we find a short vector in the lattice $[B|B_{f_1}|B_{f_2}]$.

Case 2. To obtain more short vectors in the lattice $[B|B_{f_1}|B_{f_2}]$, the attacker requests a key for small elements $k_1 f_1$ and $k_2 f_2$ for some $k_1, k_2 \in Z_p$. By the construction of GSW [5] and ABE [8], we have the following equations which we will prove a little bit later.

Lemma 2. $B_{k_1 f_1} = k_1 B_{f_1}, B_{k_2 f_2} = k_2 B_{f_2}$.

With this lemma, the attacker can get:

$$[B|B_{k_1 f_1}] \begin{bmatrix} r_1' \\ r_2' \end{bmatrix} = p(\mod q)$$

$$[B|B_{k_2 f_2}] \begin{bmatrix} u_1' \\ u_2' \end{bmatrix} = p(\mod q)$$

$$[B|B_{f_1}] \begin{bmatrix} r_1' \\ k_1 r_2' \end{bmatrix} = p(\mod q)$$

$$[B|B_{f_2}] \begin{bmatrix} u_1' \\ k_2 u_2' \end{bmatrix} = p(\mod q).$$

Hence,

$$[B|B_{f_1}|B_{f_2}] \begin{bmatrix} r_1' - u_1' \\ k_1 r_2' \\ -k_2 u_2' \end{bmatrix} = 0(\mod q).$$

It is easily to see that this results in a new short vector in the same lattice that is independent of result in the first case.

Case 3. More generally, by querying multiple functions $g_i = a_i f_1 + b_i f_2$ for $i \in [Q]$ where $a_i, b_i \in Z_p$ are small

and Q is some polynomial, the attack obtains 1-keys $[v_{1i}, v_{2i}]$ which gives the following equation:

$$\left[B|B_{g_i}\right]\begin{bmatrix}v_{1i}\\v_{2i}\end{bmatrix} = p(\mathrm{mod}q).$$

By the construction of GSW [5] and ABE [8], we have the following equations which we will prove a little bit later.

Lemma 3. *$B_{g_i} = a_i B_{f_1} + b_i B_{f_2}$ for all $i \in [Q]$*

With this lemma, we have:

$$\begin{aligned}&\left[B|B_{g_i}\right]\begin{bmatrix}v_{1i}\\v_{2i}\end{bmatrix}\\&= Bv_{1i} + B_{g_i}v_{2i}\\&= Bv_{1i} + (a_iB_{f_1} + b_iB_{f_2})v_{2i}\\&= Bv_{1i} + B_{f_1}(a_iv_{2i}) + B_{f_2}(b_iv_{2i})\\&= \left[B|B_{f_1}|B_{f_2}\right]\begin{bmatrix}v_{1i}\\a_iv_{2i}\\b_iv_{2i}\end{bmatrix}\\&= p(\mathrm{mod}q).\end{aligned}$$

Therefore, for some $i, j \in [Q]$ we have

$$\left[B|B_{f_1}|B_{f_2}\right]\begin{bmatrix}v_{1i}\\a_iv_{2i}\\b_iv_{2i}\end{bmatrix} = p(\mathrm{mod}q)$$

$$\left[B|B_{f_1}|B_{f_2}\right]\begin{bmatrix}v_{1j}\\a_jv_{2j}\\b_jv_{2j}\end{bmatrix} = p(\mathrm{mod}q).$$

Hence,

$$\left[B|B_{f_1}|B_{f_2}\right]\begin{bmatrix}v_{1i}-v_{1j}\\a_iv_{2i}-a_jv_{2j}\\b_iv_{2i}-b_jv_{2j}\end{bmatrix} = 0(\mathrm{mod}q).$$

Thus, an attacker may get a short basis for the lattice $\left[B|B_{f_1}|B_{f_2}\right]$. Since $f_1(x) = 0, f_2(x) = 0$, by computing the legitimate decryption equations he/she obtains:

$$\begin{aligned}&[B^Ts + \eta|B_{f_1}^Ts + \eta_{f_1}|B_{f_2}^Ts + \eta_{f_2}]\\&= [B|B_{f_1}|B_{f_2}]^T + noise.\end{aligned}$$

Now, the attacker may use the basis to recover the secret vector s, and hence break the security of the LWE samples that encode the attributes x.

Proof of Lemma 2: By the construction in [11], the computing process of B_f is located in the phase of *KeyGen*. Given a circuit computing a function $f : \{0,1\}^l \to \{0,1\}$, we need to conduct the following two steps in order to get B_f:

- Run the GSW Evaluation algorithm $GSW.Eval(f, \cdot)$ and then make a little change in the output phase to get the circuit corresponding to function $\hat{f} : \Psi \to \overline{\Psi}_f$.
- With the public parameters $B_1, \cdots, B_L$, run the matrices evolution algorithm $EvalF$ to compute $B_f = EvalF(B_1, \cdots, B_L, \hat{f})$.

Therefore, in order to prove the homomorphic relationship in Lemma 4.1, we only need to prove the following two homomorphic properties:

Claim 4.1. $(\hat{kf}) = k(\hat{f})$

Claim 4.2. $B_{kf} = kB_f$

Proof of Claim 4.1: Note that function $\hat{f}$ is computed from f through running the GSW evaluation algorithm $GSW.Eval(f, \cdot)$. Hence, to prove the relationship in Claim 4.1 means to prove that the GSW evaluation algorithm $GSW.Eval(f, \cdot)$ has the following homomorphic property:

$$GSW.Eval((kf), \cdot) = k \times GSW.Eval(f, \cdot).$$

Case 1. when the circuit computing f is only an addition gate, i.e. $f = x_1 + x_2$, for any GSW ciphertexts $C_1 = \begin{bmatrix}B_1\\s^TB_1 + e_1^T\end{bmatrix} + \mu_1 G, C_2 = \begin{bmatrix}B_2\\s^TB_2 + e_2^T\end{bmatrix} + \mu_2 G$, we have

$$\begin{aligned}&GSW.Eval(kf, C_1, C_2)\\&= kC_1 + kC_2\\&= \begin{bmatrix}kB_1 + kB_2\\(ks^TB_1 + ks^TB_2) + (ke_1^T + ke_2^T)\end{bmatrix} + (k\mu_1 G + k\mu_2 G)\\&= k(\begin{bmatrix}B_1 + B_2\\(s^TB_1 + s^TB_2) + (e_1^T + e_2^T)\end{bmatrix}) + k(\mu_1 G + \mu_2 G)\\&= k \cdot GSW.Eval(f, C_1, C_2).\end{aligned}$$

Case 2. when the circuit computing f is only a multiplication gate, i.e. $f = x_1 \cdot x_2$, for any GSW ciphertexts $C_1 = \begin{bmatrix}B_1\\s^TB_1 + e_1^T\end{bmatrix} + \mu_1 G, C_2 = \begin{bmatrix}B_2\\s^TB_2 + e_2^T\end{bmatrix} + \mu_2 G$, we have

$$\begin{aligned}&GSW.Eval(kf, C_1, C_2)\\&= (kC_1) \cdot G^{-1}(C_2)\\&= (\begin{bmatrix}kB_1\\ks^TB_1 + ke_1^T\end{bmatrix} + k\mu_1 G) \cdot G^{-1}(C_2)\\&= \begin{bmatrix}kB_1G^{-1}(C_2)\\ks^TB_1G^{-1}(C_2) + ke_1^TG^{-1}(C_2)\end{bmatrix} + k\mu_1 C_2\\&= \begin{bmatrix}kB_1G^{-1}(C_2) + k\mu_1 B_2\\s^T(kB_1G^{-1}(C_2) + k\mu_1 B_2) + ke_1^TG^{-1}(C_2) + k\mu_1 e_2^T\end{bmatrix}\\&\quad + k\mu_1\mu_2 G\\&= k \cdot GSW.Eval(f, C_1, C_2).\end{aligned}$$

In general, any depth d circuit can be implemented by some addition and multiplication gates, hence this homomorphic property is naturally conserved in the case of general circuits.

Proof of Claim 4.2: Note that matrix B_f is computed from $\hat{f}$ through running the matrices evolution algorithm $EvalF(B_1, \cdots, B_L, \hat{f})$. Hence, to prove the relationship in Claim 4.2 means to prove that the matrices evolution algorithm $EvalF()$ has the following homomorphic property:

$$EvalF(B_1, \cdots, B_L, k \cdot \hat{f}) = k \cdot EvalF(B_1, \cdots, B_L, \hat{f}).$$

Case 1. when the circuit computing $\hat{f}$ is only an addition gate, i.e. $f = x_1 + \cdots + x_L$, for any GSW ciphertexts $B_1, \cdots, B_L$, we have

$$\begin{aligned} &EvalF(B_1, \cdots, B_L, k \cdot \hat{f}) \\ &= [kE, \cdots, kE]^T \\ &= k[E, \cdots, E]^T \\ &= k \cdot EvalF(B_1, \cdots, B_L, \hat{f}). \end{aligned}$$

Case 2. when the circuit computing $\hat{f}$ is only an multiplication gate, i.e. $f = x_1 \times \cdots \times x_L$, for any GSW ciphertexts $B_1, \cdots, B_L$, we have

$$\begin{aligned} &EvalF(B_1, \cdots, B_L, k \cdot \hat{f}) \\ &= [O, \cdots, O, kG^{-1}(\cdots G^{-1}(-B_2 G^{-1}(-B_1)))]^T \\ &= k \cdot [O, \cdots, O, G^{-1}(\cdots G^{-1}(-B_2 G^{-1}(-B_1)))]^T \\ &= k \cdot EvalF(B_1, \cdots, B_L, \hat{f}). \end{aligned}$$

In general, any depth d circuit can be implemented by some addition and multiplication gates, hence this homomorphic property is naturally conserved in the case of general circuits.

Proof of Lemma 3: Similar to the proof of Lemma 2, here we omit it.

5 Attack #II

In this section,we provide another attack to demonstrate that the predicate encryption scheme reviewed in section 3 is insecure against an adversary that requests 1-keys. This attack exploits two types of linear error growth in the construction of the scheme in [11]. One type of this error growth is from the ciphertexts homomorphic evolution algorithm in [8]; the other one results from the GSW evaluation algorithm in [5]. Concretely, we first recall the correctness of the scheme in [11] as follows:

$$\begin{aligned} &c_{out} - \left[c_{in}^T | c_{\hat{f}}^T\right] \cdot sk_f \\ &= c_{out} - \left[c_{in}^T | \left[c_1^T | \cdots | c_L^T\right] \cdot H_{\hat{f},\Psi} + \underline{\Psi}_f\right] \cdot sk_f \\ &= c_{out} - \left[c_{in}^T | [s^T[B_1 - \psi_1 \overline{G}] + e_1^T | \cdots | s^T[B_L - \psi_l \overline{G}] \right. \\ &\quad \left. + e_L^T] \cdot H_{\hat{f},\Psi} + \underline{\Psi}_f\right] \cdot sk_f \\ &= c_{out} - \left[c_{in}^T | s^T \underbrace{[B_1 - \psi_1 \overline{G}] | \cdots | B_L - \psi_l \overline{G}] \cdot H_{\hat{f},\Psi}}_{B_f - \overline{\Psi}_f} \right. \\ &\quad \left. + \underbrace{[e_1^T | \cdots | e_L^T] \cdot H_{\hat{f},\Psi}}_{e_{ABE}} + \underline{\Psi}_f\right] \cdot sk_f \\ &= c_{out} - \left[s^T B + e_0 | s^T[B_f - \overline{\Psi}_f] + \underline{\Psi}_f + e_{ABE}\right] \cdot sk_f \\ &= c_{out} - s^T[B|B_f] \cdot sk_f - [O|(-s^T, 1)\Psi_f] \cdot sk_f \\ &\quad - [e_0 | e_{ABE}] \cdot sk_f \\ &= s^T\left[p - [B|B_f] \cdot sk_f\right] - [O|(-s^T, 1)\Psi_f] \cdot sk_f + e' \\ &\quad - [e_0 | e_{ABE}] \cdot sk_f + \lfloor \frac{q}{2} \rfloor \cdot \mu \\ &= s^T\left[p - [B|B_f] \cdot sk_f\right] - [O|f(x) \cdot (-s^T, 1)G] \cdot sk_f + e' \\ &\quad - [e_0 | e_{GSW} + e_{ABE}] \cdot sk_f + \lfloor \frac{q}{2} \rfloor \cdot \mu \\ &= e' - [e_0 | e_{GSW} + e_{ABE}] \cdot sk_f + \lfloor \frac{q}{2} \rfloor \cdot \mu, \end{aligned}$$

where the fourth equality is because of the key relation, and the final equality is because the queries requsted by adversary is 1-keys.

Note that the key sk_f is known by adversary, and by the cipthertext evolution algorithm $EvalFX$, we have

$$e_{ABE} = [e_1^T | \cdots | e_L^T] \cdot H_{\hat{f},\Psi}$$

where $H_{\hat{f},\Psi}$ can also be computed by adversary from f and Ψ through the cipthertext evolution algorithm $EvalFX$. Thus, the term e_{ABE} is linear in these original errors $e_1^T, \cdots, e_L^T$ with public coefficients.

On the other hand, by the construction of the GSW homomorphic evaluation algorithm, the term e_{GSW} is also publicly linear in the errors $e^T R_1, \cdots, e^T R_l$ which are used in the construction of the GSW fresh cipthertext Ψ.

According to the analysis above, it is not difficult to see that a single 1-key (even if it corresponds to a nonlinear function) yields a system of m linear equations in the $(l+L+2)m$ variables $e', e_0, e_1, \cdots, e_L, \hat{e}_1, \cdots, \hat{e}_l$ where $\hat{e}_1, \cdots, \hat{e}_l$ denots $R_1^T e, \cdots, R_l^T e$ respectively. By requesting $l + L + 2$ keys totally, the adversary can completely recover the above error terms, which in turn lead to recovery of the main secret s, which then permit to recover all the private attributes completely.

6 Conclusion and Open Problems

In this paper, we propose two practical attacks that demonstrate the predicate encryption scheme proposed by Brakerski *etc.* is insecure under the full attribute-hiding secrity model. The first type of attack mainly exploits two homomorphic properties in construction of the scheme; the other one, however, takes advantage of two types of linear properties in the process of error growth in the construction. This leaves open two possibilities:

1) Optimize the construction of the scheme to resist these two types of attack;

2) Look for new construction of the predicate scheme from lattice based assumptions to bypass those weak properties.

Acknowledgments

This work was supported by the Doctoral Fund of University of Jinan (Granted No. XBS1455), and National Science Foundation of Shandong Province (No. ZR2018LF006).

References

[1] H. Abdalla, X. Hu, A. Wahaballa, A. Abdalla, M. Ramadan, and Z. Qin, "Integrating the functional encryption and proxy re-cryptography to secure drm scheme," *International Journal of Network Security*, vol. 19, no. 1, pp. 27–38, 2017.

[2] S. Agrawal, "Stronger security for reusable garbled circuits, general definitions and attacks," in *Proceedings of the 37th International Cryptology Conference (CRYPTO'17)*, pp. 3–35, Santa Barbara, USA, August 2017.

[3] S. Agrawal, D. Boneh, and X. Boyen, "Lattice basis delegation in xed dimension and shorterciphertext hierarchical IBE," in *Proceedings of the 30th International Cryptology Conference (CRYPTO'10)*, pp. 98–115, Santa Barbara, USA, August 2010.

[4] S. Agrawal, D. M. Freeman, and V. Vaikuntanathan, "Functional encryption for inner product predicates from learning with errors," in *Proceedings of 17th International Conference on the Theory and Application of Cryptology and Information Security (ASIACRYPT'11)*, pp. 21–40, Seoul, Korea, December 2011.

[5] J. Alperin-Sheriff and C. Peikert, "Faster bootstrapping with polynomial error," in *Proceedings of the 30th International Cryptology Conference (CRYPTO'14)*, LNCS 8616m pp. 297–314, Springer, Aug. 2014.

[6] D. Apon, X. Fan, and F. H. Liu, "Compact identity based encryption from LWE," in *Cryptology ePrint Archive*, Report 2016/125, 2016. (`http://eprint.iacr.org/2016/125`)

[7] M. Bayat, M. Aref, "An attribute based key agreement protocol resilient to KCI attack," *International Journal of Electronics and Information Engineering*, vol. 2, no. 1, pp. 10–20, 2015.

[8] D. Boneh, C. Gentry, S. Gorbunov, S. Halevi, V. Nikolaenko, G. Segev, V. Vaikuntanathan, and D. Vinayagamurthy, "Fully key-homomorphic encryption, arithmetic circuit abe and compact garbled circuits," in *Proceedings of 33rd Annual International Conference on the Theory and Applications of Cryptographic Techniques(EUROCRYPT'14)*, pp. 533–556, Copenhagen, Denmark, May 2014.

[9] D. Boneh, A. Sahai, and B. Waters, "Functional encryption: Definitions and challenges," in *Proceedings of The 8th Theory of Cryptography Conference (TCC'11)*, pp. 253–273, Rhode Island, USA, March 2011.

[10] D. Boneh and B. Waters, "Functional encryption: Definitions and challenges," in *Proceedings of The 4th Theory of Cryptography Conference (TCC 2007)*, pp. 535–554, Amsterdam, The Netherlands, February 2007.

[11] Z. Brakerski, R. Tsabary, V. Vaikuntanathan, and H. Wee, "Private constrained prfs (and more) from lwe," in *Proceedings of The 15th Theory of Cryptography Conference (TCC'17)*, pp. 264–302, Baltimore, USA, November 2017.

[12] Z. Brakerski and V. Vaikuntanathan, "Latticebased fhe as secure as PKE," in *Proceedings of The 3rd International Conference on Information Technology and Computer Science (ITCS'14)*, pp. 1–12, Saipan, USA, July 2014.

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

[14] Z. Cao, L. Liu, 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.

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

[16] S. Garg, C. Gentry, S. Halevi, M. Raykova, A. Sahai, and B. Waters, "Candidate indistinguishability obfuscation and functional encryption for all circuits," in *Proceedings of The 54th Annual Symposium on Foundations of Computer Science (FOCS'13)*, pp. 40–49, Berkeley, California, October 2013.

[17] C. Gentry, "Fully homomorphic encryption using ideal lattices," in *Proceedings of The 41st Annual ACM Symposium on Theory of Computing (STOC'09)*, pp. 169–178, Bethesda, MD, USA, May/June 2009.

[18] C. Gentry, A. Sahai, and B. Waters, "Homomorphic encryption from learning with errors: Conceptually-simpler, asymptotically-faster, attribute-based," in *Proceedings of the 33th International Cryptology Conference (CRYPTO'13)*, pp. 75–92, Santa Barbara, USA, August 2013.

[19] S. Gorbunov, V. Vaikuntanathan, and H. Wee, "Attribute-based encryption for circuits," in *Proceedings of The 45st Annual ACM Symposium on Theory of Computing (STOC'13)*, pp. 545–554, Palo Alto, CA, USA, June 2013.

[20] S. Gorbunov, V. Vaikuntanathan, and H. Wee, "Predicate encryption for circuits from lwe," in *Proceedings of the 35th International Cryptology Conference (CRYPTO'15)*, pp. 503–523, Santa Barbara, USA, August 2015.

[21] P. Hu and H. Gao, "A key-policy attribute-based encryption scheme for general circuit from bilinear maps," *International Journal of Network Security*, vol. 19, no. 5, pp. 704–710, 2017.

[22] J. Katz, A. Sahai, and B. Waters, "Predicate encryption supporting disjunctions, polynomial equations, and inner products," in *Proceedings of 27rd Annual International Conference on the Theory and Applications of Cryptographic Techniques(EUROCRYPT'08)*, pp. 146–162, Istanbul, Turkey, April 2008.

[23] D. Micciancio and C. Peikert, "Trapdoors for lattices: Simpler, tighter, faster, smaller," in *Proceedings of 31rd Annual International Conference on the Theory and Applications of Cryptographic Techniques(EUROCRYPT'12)*, pp. 700–718, Cambridge, United Kingdom, April 2012.

[24] Y. Wang, "Lattice ciphertext policy attribute-based encryption in the standard model," *International Journal of Network Security*, vol. 16, no. 6, pp. 444–451, 2014.

[25] L. Zhang and H. Yin, "Recipient anonymous ciphertext-policy attribute-based broadcast encryption," *International Journal of Network Security*, vol. 20, no. 1, pp. 168–176, 2018.

Biography

Chengbo Xu received the B.S. degree in Mathematics from the Liaocheng University, China, in 2002, the M.S. degree in Cryptology from the Hubei University, China, in 2005, and the Ph.D. degree in Computer Science from the Beijing University of Post and Telecommunication, China, in 2014, respectively. Currently, He is a Lecture in the School of Mathimatical Sciences at University of Jinan. His research interests include information security and cryptology. Dr. Xu may be reached at cbqysy@163.com.

Research on Batch Verification Schemes for Identifying Illegal Signatures

Hsieh-Tsen Pan[1], Eko Fajar Cahyadi[1,2], Shu-Fen Chiou[3], and Min-Shiang Hwang[1,4]
(Corresponding author: Min-Shiang Hwang)
Department of Computer Science & Information Engineering, Asia University, Taichung, Taiwan[1]
Department of Telecommunication Engineering, Institut Teknologi Telkom Purwokerto, Purwokerto, Indonesia[2]
Department of Information Management, National Taichung University of Science and Technology, Taiwan[3]
Department of Medical Research, China Medical University Hospital, China Medical University, Taiwan[4]
(Email: mshwang@asia.edu.tw)
(Received Mar. 3, 2019; revised and accepted Oct. 1, 2019)

Abstract

Invalid signatures produced by some adversaries may pose a severe challenge to the recipient. Furthermore, identifying an invalid signature in a bunch of messages could be a complex and challenging task to do. Batch verification is an idea to simultaneously verify multiple digital signatures in just one exponential operation time. By this scheme, we can make a quick response, and improve the verification time. In terms of identifying illegal signatures in the batch of messages, we have surveyed several well-known papers that proposed different approaches. In this paper, we define four criteria for evaluating these schemes. It is followed by a detailing review and computation comparisons from all documents. Finally, we provide two issues for future works.

Keywords: Batch Verification Schemes; Digital Signature; Identifying Illegal Signatures

1 Introduction

A digital signature is a method for signing a transmitted electronic document so that the other parties can verify its contents and the sender's identity. Each signer has a pair of keys: a private key and a public key. The private key is kept being secret, while the public key is made public. A signer creates a digital signature using their private key, while a recipient can verify the digital signature by the signer's public key. No one can forge the signer's digital signature as the private key is safely guarded [6, 11, 18].

If a signer wants to generate and send t signatures, then the verifier needs to check for t signatures. It is inefficient since they should spend t times to validate t digital signatures using the traditional cryptosystems. The batch verification schemes were proposed to verify these multiple digital signatures by the signer's public key, which needs only one verification instead of t verifications [5, 7, 9, 13, 15, 17]. However, if the batch verification fails, all the individual signatures must be verified separately, which would become inefficient.

In this survey, we provide a relational approach of several representative publications that have a common interest in batch verification schemes for illegal signatures identification [1, 12, 14, 16]. A detailed overview of those researches is focused on their strengths and weaknesses.

To achieve effectivity and efficiency evaluation of identifying illegal signatures in a batch verification scheme, we propose the following four criteria.

1) Unforgeability: No one can forge the legitimate multiple digital signatures. It's the basic requirement of the batch verification scheme.

2) Efficient traceability: It should be able to determine illegal signatures efficiently. When the multiple digital signatures are forged, the verifier can identify them with minimum computational and communication costs.

3) Applicability: A generic illegal signature locating algorithm could be used in any type of batch verification scheme. Since the effort is reinvested over a larger number of applications, the single generic illegal signature locating algorithm may be optimized, verified, and otherwise improved [**?**].

4) Error locating auditability: It does not misjudge the legal signature as an illegal signature.

For a better understanding, the rest of this paper is organized as follows. Section 2 describes various batch verification schemes for identifying illegal signatures. In Section 3, we give an analysis and a comparison among these schemes. In Section 4, two issues for future works are proposed. Finally, the conclusion is explained in Section 5.

2 Related Works

Several batch verification schemes for identifying illegal signatures have been proposed [1, 12, 14, 16].

2.1 A GCD-based Batch Verification Scheme for Identifying Illegal Signatures

In 2002, Hwang, Lee, and Lai proposed a batch verification scheme for identifying illegal signatures [12]. Their scheme is based on the Greatest Common Divisor (GCD). If a signer Alice wants to transmit the message M to receiver Bob, she must generate a digital signature S by using two assumptions: One is that $\prod_{i=1}^{t} h(M_i) < n$, and the other is $h(M_i)$, must be a prime, where $h(.)$ represents a public one-way hashing function, and $i = 1, 2, \cdots, t$. Once the signer sends of t message and signature pairs $((M_i, S_i), i = 1, 2, \cdots, t)$, the verifier will perform the following procedures to authenticate the illegal signature.

Step 1: The verifier computes $A = \gcd[((\prod_{i=1}^{t} S_i)^e \bmod n), \prod_{i=1}^{t} h(M_i)]$.

Step 2: The verifier computes $B = (\prod_{i=1}^{t} S_i)^e \bmod n/A$. If $B = 1$, these signatures are legal. Otherwise, one or more signatures are illegal.

Step 3: If B is a prime, the message (M'_j, S'_j) is illegal. Otherwise, check the following

$$B \bmod M'_j \stackrel{?}{=} 0, j = 1, 2, \cdots, t.$$

If the above equation holds, the message (M'_j, S'_j) is illegal.

If there is only one illegal signature $(S'_j, j \in (1, 2, \cdots, t))$ hiding among the t signatures, Hwang-Lee-Lai's scheme is efficient. However, it needs $t-1$ modulus remainder operations to identify these illegal signatures in Hwang-Lee-Lai's scheme.

For example, suppose the signer sends 5 messages $((M_1, S_1), (M_2, S_2), (M_3, S_3), (M_4, S_4), (M_5, S_5))$ to the verifier. If there is one illegal signature (S'_4) among the five signatures, it would be identified as follows:

$$\begin{aligned} A &= \gcd[((\prod_{i=1}^{5} S_i)^e \bmod n), \prod_{i=1}^{5} h(M_i)] \\ &= h(M_1)h(M_2)h(M_3)h(M_5). \\ B &= (S_1S_2S_3S'_4S_5)^e \bmod n/A \\ &= \frac{h(M_1)h(M_2)h(M_3)h(M'_4)h(M_5)}{h(M_1)h(M_2)h(M_3)h(M_5)} \\ &= h(M'_4) \end{aligned}$$

Thus, the illegal message is M'_4. However, if there are two illegal signatures, S'_2 and S'_4, among the five, it will not be directly identified because of,

$$\begin{aligned} A &= \gcd[((\prod_{i=1}^{5} S_i)^e \bmod n), \prod_{i=1}^{5} h(M_i)] \\ &= h(M_1)h(M_3)h(M_5). \\ B &= (S_1S'_2S_3S'_4S_5)^e \bmod n/A \\ &= \frac{h(M_1)h(M'_2)h(M_3)h(M'_4)h(M_5)}{h(M_1)h(M_3)h(M_5)} \\ &= h(M'_2)h(M'_4). \end{aligned}$$

The verifier cannot directly identify the illegal signatures from the multiplication of $h(M'_2)h(M'_4)$. The verifier needs to identify invalid signatures as follows:

$$\begin{aligned} B \bmod M'_1 &\neq 0 \\ B \bmod M'_2 &= 0 \\ B \bmod M'_3 &\neq 0 \\ B \bmod M'_4 &= 0 \\ B \bmod M'_5 &\neq 0. \end{aligned}$$

From the above equations, the verifier identifies two illegal messages: $(M'_2, S'_2), (M'_4, S'_4)$.

2.2 A 2D-based Batch Verification Scheme for Identifying Illegal Signatures

In 2010, a 2D-based batch verification scheme for identifying illegal signatures was proposed by Li, Hwang, and Chen [14]. When the verifier receives the messages $(M_1, S_1), (M_2, S_2), \cdots, (M_t, S_t)$ from the signer, the verifier will generate an $m \times n$ matrix, where m is the smallest integer which satisfies $m \times n \geq t$, where t is random numbers. The verifier performs the following procedures to verify the illegal signature.

Step 1: The verifier constructs an $m \times n$ matrix (see Table 1).

Table 1: An $m \times m$ matrix

S(1,1)	S(1,2)	...	S(1,m-1)	S(1,m)
S(2,1)	S(2,2)	...	S(2,m-1)	S(2,m)
⋮	⋮	⋮	⋮	⋮
S(m-1,1)	S(m-1,2)	...	S(m-1,m-1)	S(m-1,m)
S(m,1)	S(m,2)	...	S(m,m-1)	S(m,m)

Step 2: The verifier randomly selects and fills these t digital signatures in the $m \times n$ matrix.

Step 3: The verifier performs the batch verify each of the rows. The details of row verifications are computed as follows:

$$(\prod_{i=1}^{m} S_{(r,i)})^e \stackrel{?}{=} \prod_{i=1}^{m} h(M_{(r,i)}) \bmod N, \ r = 1, 2, \cdots, m).$$

Step 4: The verifier performs the batch verify each of the columns. The details of column verifications are computed as follows:

$$(\prod_{i=1}^{m} S_{(i,c)})^e \stackrel{?}{=} \prod_{i=1}^{m} h(M_{(i,c)}) \bmod N,\ c = 1, 2, \cdots, m).$$

Step 5: If there are some signature-verification faults in the matrix, the verifier could find out where these signature-verification faults are located by finding the matrix positions of row and column overlaps.

For example, suppose the signer sends 16 messages $((M_1, S_1), (M_2, S_2), \cdots, (M_{16}, S_{16}))$ to the verifier. The verifier will generate 36 signatures by random selections and fills these signatures in the 4×4 matrix (see Table 2).

Table 2: A 4×4 matrix

S(1,1)	S(1,2)	S(1,3)	S(1,4)
S(2,1)	S(2,2)	S(2,3)	S(2,4)
S(3,1)	S(3,2)	S(3,3)	S(3,4)
S(4,1)	S(4,2)	S(4,3)	S(4,4)

Assume there is one illegal signature in the position $S(2, 3)$ of matrix. There would occur two verification failures in the second row and the third column, respectively (see Table 3).

Table 3: A 4×4 matrix with one illegal signature

S(1,1)	S(1,2)	S(1,3)	S(1,4)	Pass
S(2,1)	S(2,2)	$S^*(2,3)$	S(2,4)	Fail
S(3,1)	S(3,2)	S(3,3)	S(3,4)	Pass
S(4,1)	S(4,2)	S(4,3)	S(4,4)	Pass
Pass	Pass	Fail	Pass	

According to the overlap of verification failures of the second row and the third column, the illegal signature could be precisely identified in the position $S(2, 3)$ of the matrix.

2.3 A BT-based Batch Verification Scheme for Identifying Illegal Signatures

In 2013, a binary tree-based (BT-based) batch verification scheme for identifying illegal signatures was proposed by Atanasiu [1]. When the verifier receives the messages $(M_1, S_1), (M_2, S_2), \cdots, (M_t, S_t)$ from the signer, the verifier will re-order these signatures by a total order relation and perform the following procedures to verify the illegal signature.

Step 1: The verifier re-orders these signatures by a total order relation: $(M'_1, S'_1), (M'_2, S'_2), \cdots, (M'_t, S'_t)$. Here, $(M'_1, S'_1) < (M'_2, S'_2) < \cdots < (M'_t, S'_t)$ are by the following rule:

$$(M'_i, S'_i) < (M'_j, S'_j) \iff (S'_i < S'_j) \vee [(S'_i = S'_j) \wedge (M'_i < M'_j)].$$

Step 2: The verifier performs one time of the batch verify with all t signatures:

$$(\prod_{i=1}^{t} S'_i)^e \stackrel{?}{=} \prod_{i=1}^{t} h(M'_i).$$

Step 3: If the above equation holds, all t signatures are legal. Otherwise, the verifier divides t signatures into two parts: $[(M'_1, S'_1), (M'_2, S'_2), \cdots, (M'_{t/2}, S'_{\lceil \frac{t}{2} \rceil})]$ and $[(M'_{\lceil \frac{t}{2}+1 \rceil}, S'_{\lceil \frac{t}{2}+1 \rceil}), (M'_{\lceil \frac{t}{2}+2 \rceil}, S'_{\lceil \frac{t}{2}+2 \rceil}), \cdots, (M'_t, S'_t)]$. Next, the verifier repeatedly performs Steps 2 and 3 for all parts.

For example, suppose the signer sends 16 signatures $((M_1, S_1), (M_2, S_2), \cdots, (M_{16}, S_{16}))$ to the verifier. The verifier re-orders these signatures by a total order relation: $(M'_1, S'_1), (M'_2, S'_2), \cdots, (M'_{16}, S'_{16})$ such that $(M'_1, S'_1) < (M'_2, S'_2) < \cdots < (M'_{16}, S'_{16})$.

Assume there is one illegal signature: S'_{15}. The verifier performs the following procedures:

Step 1: The verifier performs one times of the batch verify with all 16 signatures:

$$(\prod_{i=1}^{16} S'_i)^e \stackrel{?}{=} \prod_{i=1}^{16} h(M'_i).$$

Since there is one illegal signature: S'_{15}, the above equation is not held. The verifier divides these 16 signatures into two parts: Part 1: $[(M'_1, S'_1), (M'_2, S'_2), \cdots, (M'_8, S'_8)]$ and Part 2: $[(M'_9, S'_9), (M'_{10}, S'_{10}), \cdots, (M'_{16}, S'_{16})]$.

Step 2: The verifier performs one time of batch verification with all signatures in Part 1:

$$(\prod_{i=1}^{8} S'_i)^e \stackrel{?}{=} \prod_{i=1}^{8} h(M'_i).$$

Since there are not illegal signatures in Part 1, the above equation is held.

Step 3: The verifier performs one time of batch verification with all signatures in Part 2:

$$(\prod_{i=9}^{16} S'_i)^e \stackrel{?}{=} \prod_{i=9}^{16} h(M'_i).$$

Since there is one illegal signature: S'_{15}, the above equation is not held. The verifier divides these 8 signatures into two parts: Part 3: $[(M'_9, S'_9), (M'_{10}, S'_{10}), \cdots, (M'_{12}, S'_{12})]$ and Part 4: $[(M'_{13}, S'_{13}), (M'_{14}, S'_{14}), \cdots, (M'_{16}, S'_{16})]$.

Step 4: The verifier performs one time of batch verification with all signatures in Part 3:

$$(\prod_{i=9}^{12} S_i')^e \stackrel{?}{=} \prod_{i=9}^{12} h(M_i').$$

Since there are not illegal signatures in Part 3, the above equation is held.

Step 5: The verifier performs one time of batch verification with all signatures in Part 4:

$$(\prod_{i=13}^{16} S_i')^e \stackrel{?}{=} \prod_{i=13}^{16} h(M_i').$$

Since there is one illegal signature: S_{15}', the above equation is not held. The verifier divides these 4 signatures into two parts: Part 5: $[(M_{13}', S_{13}'), (M_{14}', S_{14}')]$ and Part 6: $[(M_{15}', S_{15}'), (M_{16}', S_{16}')]$.

Step 6: The verifier performs one time of batch verification with all signatures in Part 5:

$$(S_{13}' S_{14}')^e \stackrel{?}{=} h(M_{13}' M_{14}').$$

Since there are not illegal signatures in Part 5, the above equation is held.

Step 7: The verifier performs one time of batch verification with all signatures in Part 6:

$$(S_{15}' S_{16}')^e \stackrel{?}{=} h(M_{15}' M_{16}').$$

Since there is one illegal signature: S_{15}', the above equation is not held. The verifier divides these 8 signatures into two parts: Part 7: $[(M_{15}', S_{15}')]$ and Part 8: $[(M_{16}', S_{16}')]$.

Step 8: The verifier performs one time of batch verification with all signatures in Part 7:

$$(S_{15}')^e \stackrel{?}{=} h(M_{15}').$$

Since there is one illegal signature: S_{15}', the above equation is not held, so the verifier knows the (M_{15}', S_{15}') is illegal.

Step 9: The verifier performs one time of batch verification with all signatures in Part 8:

$$(S_{16}')^e \stackrel{?}{=} h(M_{16}').$$

Since there are not illegal signatures in Part 8, the above equation is held.

2.4 An n-Dimension-based Batch Verification Scheme for Identifying Illegal Signatures

In 2015, an n-Dimension-based batch verification scheme for identifying illegal signatures was proposed by Ren *et al.* [16]. Their approach is an extended version of a 2D-based batch verification scheme for identifying invalid signatures [14]. For the sake of understanding their method, we introduce the 3D-based batch verification scheme for identifying illegal signatures ($n = 3$).

When the verifier receives the messages $(M_1, S_1), (M_2, S_2), \cdots, (M_t, S_t)$ from the signer, the verifier will generate an $m \times m \times m$ matrix in which m is the smallest integer which satisfies $m^3 \geq t$. The verifier performs the following procedures to verify the illegal signature.

Step 1: The verifier constructs an $m \times m \times m$ matrix.

Step 2: The verifier randomly selects and fills these t digital signatures in the $m \times m \times m$ matrix.

Step 3: The verifier performs the batch verification of each plane. The details of x-axis plane verifications are computed as follows:

$$(\prod_{i=0}^{m-1}\prod_{j=0}^{m-1} S_{(x,i,j)})^e \stackrel{?}{=} \prod_{i=0}^{m-1}\prod_{j=0}^{m-1} h(M_{(x,i,j)}),$$
$$x = 0, 1, \cdots, (m-1).$$

Step 4: The verifier performs the batch verification of each plane. The details of y-axis plane verifications are computed as follows:

$$(\prod_{i=0}^{m-1}\prod_{j=0}^{m-1} S_{(i,y,j)})^e \stackrel{?}{=} \prod_{i=0}^{m-1}\prod_{j=0}^{m-1} h(M_{(i,y,j)}),$$
$$y = 0, 1, \cdots, (m-1).$$

Step 5: The verifier performs the batch verification of each plane. The details of z-axis plane verifications are computed as follows:

$$(\prod_{i=0}^{m-1}\prod_{j=0}^{m-1} S_{(i,j,z)})^e \stackrel{?}{=} \prod_{i=0}^{m-1}\prod_{j=0}^{m-1} h(M_{(i,j,z)}),$$
$$z = 0, 1, \cdots, (m-1).$$

Step 6: If there are some signature-verification faults in the matrix, the verifier could find out where these signature-verification faults are located by finding the matrix positions of x, y, and z-axis plane overlap.

3 Comparisons

In this section, we compare these schemes introduced in Section 2 in terms of efficiency, the type of batch verification scheme (BVS) and misidentification (see Table 4).

3.1 Analysis of Hwang-Lee-Lai's Scheme

In Hwang-Lee-Lai's batch verification scheme for identifying illegal signatures [12], there are two assumptions:

Table 4: Comparisons among the batch verification schemes for identifying illegal signatures

Schemes	Computations with illegal signature	Computations with two or more illegal signatures	The Type of Batch Verification Scheme (BVS)	Misidentification
Hwang-Lee-Lai's Scheme [12]	$t/10E$	$t/10E + (t-1)MR$	RSA-Type BVS	No
Li-Hwang-Chen's Scheme [14]	$2\lceil\sqrt{t}\rceil$	$2\lceil\sqrt{t}\rceil$	Any Types	Yes
Atanasiu's Scheme [1]	$1+\log t$	$\frac{(1+2\log t)+(2\log t+2\log\lceil\frac{t}{2}\rceil)}{2}$	Any Types	No
Ren *et al.*'s scheme [16]	$n\lceil\sqrt[n]{t}\rceil$	$n\lceil\sqrt[n]{t}\rceil$	Any Types	Yes

t : The number of digital signatures
E : Exponential operations
MR : Modulus Remainder operations

One is that $\prod_{i=1}^{t} h(M_i) < n$, and the other is that $h(M_i)$, must be a prime where $i = 1, 2, \cdots, t$.

In a secure digital signature, the length of the signature is 1024 bits. Thus, for satisfying the assumption, $\prod_{i=1}^{t} h(M_i) < n$, the length of $h(M_i)$ is selected as 10 bits. This means the maximus of $h(M_i)$ is 1,021 which is the maximal prime of less than the $2^{10} = 1,024$. Therefore, the total computations for identifying illegal signatures in Hwang-Lee-Lai's scheme needs $t/10$ exponential operations and $t-1$ modulus remainder operations.

In Hwang-Lee-Lai's scheme, the verifier needs to verify by RSA-type batch verification scheme (BVS). Their method does not misjudge the legal signature as an illegal signature.

3.2 Analysis of Li-Hwang-Chen's Scheme

In Li-Hwang-Chen's batch verification scheme for identifying illegal signatures [14], the verifier needs to constructs an $m \times n$ matrix, where m and n are the smallest integer that satisfies $m \times n \geq t$.

In Li-Hwang-Chen's scheme, the verifier needs to verify each row and column by a general batch verification scheme [2–4, 7, 8, 10]. Any multiple digital signature schemes could be used in Li-Hwang-Chen's scheme.

For the sake of comparison, the batch verification scheme of their scheme is used in RSA signature. The computation of the RSA batch verification scheme is one exponential operation (one time verification). Therefore, the total computations for identifying illegal signatures in Li-Hwang-Chen's scheme needs $2\lceil\sqrt{t}\rceil$ exponential operations.

In Li-Hwang-Chen's scheme, the illegal signature could be precisely identified if it is only one illegal signature. However, it will have misidentification if two or more illegal signatures occur. For example, assume there are two illegal signatures in the positions $S(4,1)$ and $S(2,3)$ of matrix in Table 5. There would occur fourth verification failures in the second and the fourth rows and the first and third columns, respectively (see Table 6). However, obviously there are only two of them are real invalid signatures. There are two legal signatures, S(2,1) and S(4,3), misidentified as illegal signatures.

Table 5: A 4×4 matrix with two illegal signatures

S(1,1)	S(1,2)	S(1,3)	S(1,4)	Pass
S(2,1)	S(2,2)	$S^*(2,3)$	S(2,4)	Fail
S(3,1)	S(3,2)	S(3,3)	S(3,4)	Pass
$S^*(4,1)$	S(4,2)	S(4,3)	S(4,4)	Fail
Fail	Pass	Fail	Pass	

Table 6: Four verification failure in a 4×4 matrix with two illegal signatures

S(1,1)	S(1,2)	S(1,3)	S(1,4)	Pass
S(2,1)	S(2,2)	$S^*(2,3)$	S(2,4)	Fail
S(3,1)	S(3,2)	S(3,3)	S(3,4)	Pass
$S^*(4,1)$	S(4,2)	S(4,3)	S(4,4)	Fail
Fail	Pass	Fail	Pass	

3.3 Analysis of Atanasiu's Scheme

In Atanasiu's batch verification scheme for identifying illegal signatures [1], the verifier needs to verify each parts by a general batch verification scheme [2–4, 7, 8, 10]. Any multiple digital signature schemes could be used in his scheme.

For the sake of comparison, the batch verification scheme of their scheme is used in RSA signature. The computation of the RSA batch verification scheme is one exponential operation (one time verification). Therefore, the total computations for identifying illegal signatures in Atanasiu's scheme needs $1+2\log t$ exponential operations if it is only one illegal signature. For example, assume there is one illegal signature: S'_{15} in Figure 1. The verifier needs to perform the following batch verification: $\{P_0$, P_1, P_2, P_5, P_6, P_{13}, P_{14}, S'_{15}, and $S_{16}\}$. The total computations for identifying illegal signatures is $1 + 2\log t = 9$ exponential operations.

The best case of the total computations for identifying illegal signatures in Atanasiu's scheme needs $1 + 2\log t$

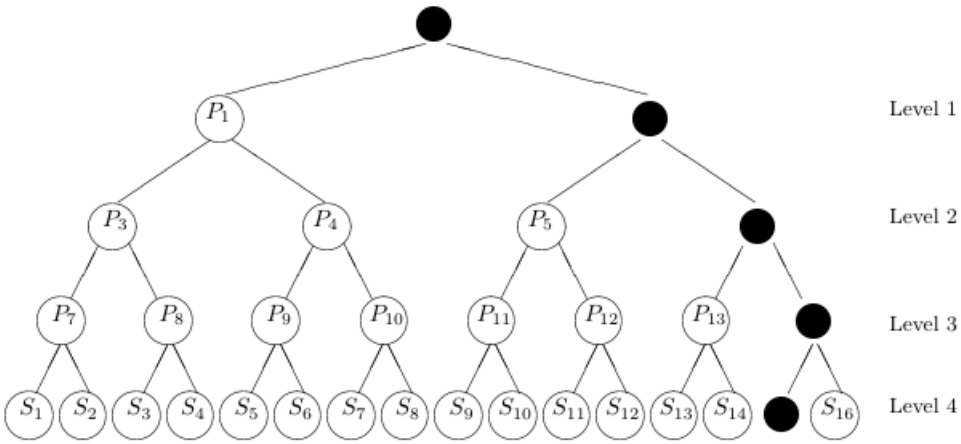


Figure 1: An example of an illegal signature: S'_{15}

exponential operations if it has two illegal signatures in the same part. For example, assume there are two illegal signatures: S'_{15} and S'_{16} in Figure 2. The verifier needs to perform the following batch verification: $\{P_0, P_1, P_2, P_5, P_6, P_{13}, P_{14}, S'_{15}$, and $S_{16}\}$. The total computations for identifying illegal signatures is $1 + 2\log t = 9$ exponential operations.

The worst case of the total computations for identifying illegal signatures in Atanasiu's scheme needs $1 + 2\log t + 2\log\lceil\frac{t}{2}\rceil$ exponential operations if it has two illegal signatures in the different parts. For example, assume there are two illegal signatures: S'_1 and S'_{15} in Figure 2. The verifier needs to perform the following batch verification: $\{(P_0, P_1, P_2, P_5, P_6, P_{13}, P_{14}, S'_{15}, S_{16})$ and $(P_3, P_4, P_7, P_8, S'_1, S_2\}$. The total computation for identifying illegal signatures is $1 + 2\log 16 + 2\log\lceil\frac{16}{2}\rceil = 15$ exponential operations.

The average computation for identifying two illegal signatures in Atanasiu's scheme needs $\frac{(1+2\log t)+(2\log t+2\log\lceil\frac{t}{2}\rceil)}{2}$ exponential operations. Obviously, the average computations for identifying two or more illegal signatures in Atanasiu's scheme needs more exponential operations than Hwang-Lee-Lai's and Li-Hwang-Chen's schemes.

The more the number of illegal signatures, the less efficient Atanasiu's scheme. For example, assume there are three illegal signatures: S'_1, S'_5, and S'_{15} in Figure 3. The verifier needs to perform the following batch verification: $\{(P_0, P_1, P_2, P_3, P_4, P_7, P_8, S'_1, S_2)$, (P_9, P_{10}, S'_5, S_6), and $P_5, P_6, P_{13}, P_{14}, S'_{15}, S_{16})\}$. The total computation for identifying illegal signatures is $9 + 4 + 6 = 19$ exponential operations.

The worst case of the total computations for identifying illegal signatures in Atanasiu's scheme needs $2t - 1$ exponential operations if it has $\frac{t}{2}$ illegal signatures in the different parts. For example, assume there are eight illegal signatures: S'_1, S'_3, S'_5, S'_7, S'_9, S'_{11}, S'_{13}, and S'_{15} in Figure 4. The verifier needs to perform the following batch verification: $\{(P_0, P_1, P_2, P_3, P_4, P_7, P_8, S'_1, S_2)$, (S'_3, S_4), (P_9, P_{10}, S'_5, S_6), (S'_7, S_8), $(P_5, P_6, P_{11}, P_{12}, S'_9, S_{10})$, (S'_{11}, S_{12}), $(P_{13}, P_{14}, S'_{13}, S_{14})$, and $(S'_{15}, S_{16})\}$. The total computation for identifying illegal signatures is $9 + 2 + 4 + 2 + 6 + 2 + 4 + 2 = 35$ exponential operations.

3.4 Analysis of Ren *et al.*'s Scheme

In Ren *et al.*'s batch verification scheme for identifying illegal signatures [16], the verifier needs to constructs an $\underbrace{m \times m \times \cdots \times m}_{n}$ n-Dimension matrix where m is the smallest integer which satisfies $m^3 \geq t$. This means the verifier could select $m = \lceil \sqrt[n]{t} \rceil$.

In Ren *et al.*'s scheme, the verifier needs to verify each plane. Any multiple digital signature schemes could be used in their scheme.

For the sake of comparison, the batch verification scheme of their scheme is used in RSA signature. The computation of the RSA batch verification scheme is one exponential operation (one time verification). Therefore, the total computation for identifying illegal signatures in Ren *et al.*'s scheme needs $n\lceil \sqrt[n]{t} \rceil$ exponential operations. For example, if there are 1000 signatures, the total computation for identifying illegal signatures in Ren *et al.*'s cheme needs $3\lceil \sqrt[3]{1000} \rceil = 30$ exponential operations.

In Ren *et al.*'s scheme, the illegal signature could be precisely identified if it is only one illegal signature. However, it will have misidentification if two or more illegal signatures occur.

4 Future Works

In this section, we propose two issues for future works.

1) An efficient batch verification scheme for identifying illegal signatures. The verifier could precisely identify these illegal signatures with low computation.

2) An application of a batch verification scheme for identifying illegal signatures. There are many appli-

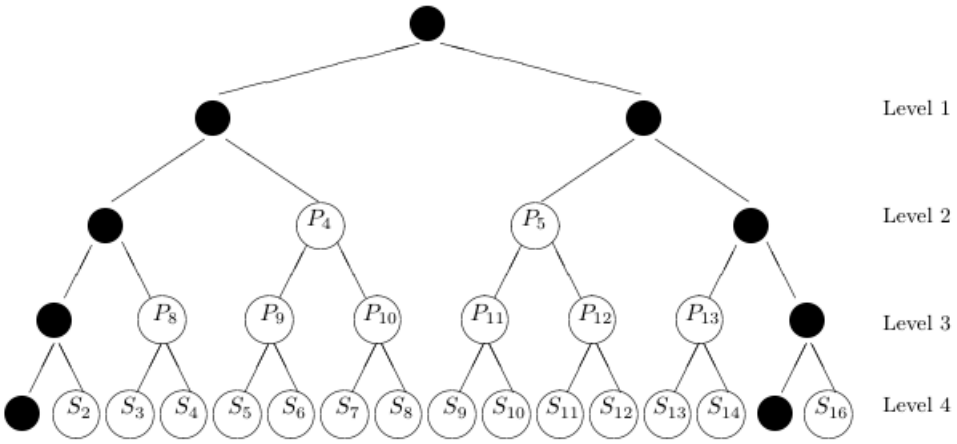


Figure 2: An example of two illegal signatures: S'_1 and S'_{15}

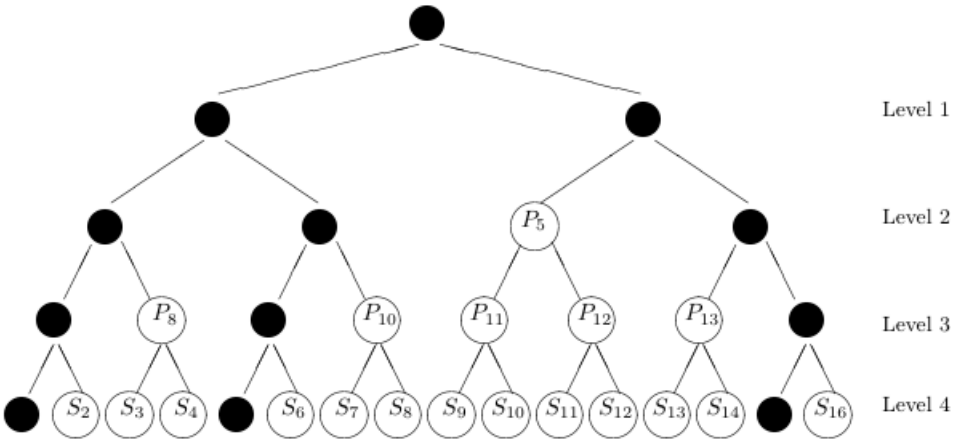


Figure 3: An example of three illegal signatures: S'_1, S'_5, and S'_{15}

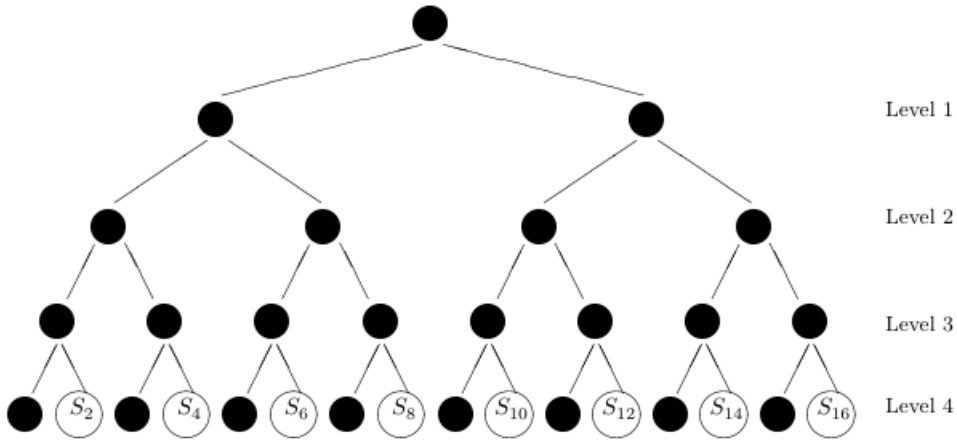


Figure 4: An example of eight illegal signatures: S'_1, S'_3, S'_5, S'_7, S'_9, S'_{11}, S'_{13}, and S'_{15}

cations in Internet of Thing (IoT). These applications with a lot of data need to verify the legal messages and signatures efficiently.

5 Conclusions

In this paper, we have reviewed some batch verification schemes for identifying illegal signatures, and proposed some criteria for evaluating these schemes. We also proposed two issues for future works.

Acknowledgments

This research was partially supported by the Ministry of Science and Technology, Taiwan (ROC), under contract no.: MOST 108-2410-H-468-023 and MOST 108-2622-8-468-001-TM1.

References

[1] A. Atanasiu, "A new batch verifying scheme for identifying illegal signatures," *Journal of Computer Science and Technology*, vol. 28, no. 1, pp. 1–8, 2013.

[2] F. Bao, C. C. Lee, M. S. Hwang, "Cryptanalysis and improvement on batch verifying multiple RSA digital signatures," *Applied Mathematics and Computation*, vol. 172, no. 2, pp. 1195–1200, 2006.

[3] T. Y. Chang, M. S. Hwang, W. P. Yang, K. C. Tsou, "A modified Ohta-Okamoto digital signature for batch verification and its multi-signature version," *International Journal of Engineering and Industries*, vol. 3, no. 3, pp. 75–83, 2012.

[4] S. W. Changchien and M. S. Hwang, "A batch verifying and detecting multiple RSA digital signatures," *International Journal of Computational and Numerical Analysis and Applications*, vol. 2, no. 3, pp. 303–307, 2002.

[5] L. Harn, "Batch verifying multiple RSA digital signatures," *Electronics Letters*, vol. 34, no. 12, pp. 1219–1220, 1998.

[6] 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.

[7] M. S. Hwang, T. Y. Chang, W. P. Yang, "The batch verifying multiple digital signature scheme: A modified version of Ohta-Okamoto digital signature," in *The 3rd International Conference on Next Generation Information Technology (ICNIT'12)*, pp. 732–735, 2012.

[8] M. S. Hwang and C. C. Lee, "Research issues and challenges for multiple digital signatures," *International Journal of Network Security*, vol. 1, no. 1, pp. 1–7, 2005.

[9] M. S. Hwang, C. C. Lee, and Eric J. L. Lu, "Cryptanalysis of the batch verifying multiple DSA-type digital signatures," *Pakistan Journal of Applied Sciences*, vol. 1, no. 3, pp. 287–288, 2001.

[10] 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.

[11] M. S. Hwang, C. C. Lee, and Y. C. Lai, "Traceability on RSA-based partially signature with low computation," *Applied Mathematics and Computation*, vol. 145, no. 2-3, pp. 465–468, 2003.

[12] M. S. Hwang, C. C. Lee, Y. C. Lai, "Detecting the illegal signature in multiple signatures," in *International Conference on Advanced Communications Technology (ICACT'02)*, pp. 881–882, 2002.

[13] M. S. Hwang, I. C. Lin, and K. F. Hwang, "Cryptanalysis of the batch verifying multiple RSA digital signatures," *Informatica*, vol. 11, no. 1, pp. 15–19, 2000.

[14] C. T. Li, M. S. Hwang, S. M. Chen, "A batch verifying and detecting the illegal signatures," *International Journal of Innovative Computing, Information and Control*, vol. 6, no. 12, pp. 5311–5320, 2010.

[15] D. Naccache, D. Mraihi, D. Rapheali, and S. Vaudenay, "Can DSA be improved: Complexity trade-offs with the digital signature standard," in *Proceedings of Eurocrypt'94*, pp. 85–94, Lecture Notes in Computer Science, 1994.

[16] Y. L. Ren, S. Wang, X. P. Zhang, and M. S. Hwang, "An efficient batch verifying scheme for detecting illegal signatures," *International Journal of Network Security*, vol. 17, no. 4, pp. 463–470, 2015.

[17] S. F. Tzeng, C. C. Lee, and M. S. Hwang, "A batch verification for multiple proxy signature," *Parallel Processing Letters*, vol. 21, no. 1, pp. 77–84, 2011.

[18] S. F. Tzeng, C. Y. Yang, and M. S. Hwang, "A new digital signature scheme based on factoring and discrete logarithms," *International Journal of Computer Mathematics*, vol. 81, no. 1, pp. 9–14, 2004.

Biography

Hsien-Tsen Pan received B.S. in Business Administration From Soochow University Taipei Taiwan in 1999; M.S in Information Engineering, Asia University Taichung Taiwan 2015; Doctoral Program of Information Engineering, Asia University Taichung Taiwan from 2015 till now. From 2011 to 2014, he was the manager in Enterprise Service Chunghwa Telecom South Branch Taichung Taiwan. From 2014 to 2017, he was the operation manager in Medium division Taiwan Ricoh Co., Ltd. Taichung Taiwan From 2017 Sep 20 he is the Apple MDM Server Service VP in Get Technology Co.Ltd. Taipei Taiwan.

Eko Fajar Cahyadi is a lecturer in the Department of Telecommunication Engineering, Institut Teknologi Telkom Purwokerto, Indonesia. He currently pursuing a Ph.D. degree in the Department of Computer Science

and Information Engineering at Asia University, Taiwan. He receives the B. Eng. and M. Sc. degree in electrical engineering from Institut Sains dan Teknologi Akprind Yogyakarta in 2009, and Institut Teknologi Bandung in 2013, respectively. His research interest includes wireless network security, optical fiber communication, and teletraffic engineering.

Shu-Fen Chiou received a B.B.A degree in Information Management from National Taichung Institute of Technology, Taichung, Taiwan, ROC, in 2004; She studied M.S. degree in Computer Science and Engineering from National Chung Hsing University for one year, and she started to pursue the Ph.D. degree. She received a Ph. D. from Computer Science and Engineering from National Chung Hsing University in 2012. She is currently an assistant professor of department of Information Management, National Taichung University of Science and Technology. Her current research interests include information security, network security, data hiding, text mining and big data analysis.

Min-Shiang Hwang received M.S. in industrial engineering from National Tsing Hua University, Taiwan in 1988; and Ph.D. degree in computer and information science from National Chiao Tung University, Taiwan in 1995. He was a professor and Chairman of the Department of Management Information Systems, NCHU, during 2003-2009. He was also a visiting professor with University of California (UC), Riverside and UC. Davis (USA) during 2009-2010. He was a distinguished professor of Department of Management Information Systems, NCHU, during 2007-2011. He obtained the 1997, 1998, 1999, 2000, and 2001 Exceilent Research Award of National Science Council (Taiwan). Dr. Hwang was a dean of College of Computer Science, Asia University (AU), Taichung, Taiwan. He is currently a chair professor with Department of Computer Science and Information Engineering, AU. His current research interests include information security, electronic commerce, database and data security, cryptography, image compression, and mobile computing. Dr. Hwang has published over 300+ articles on the above research fields in international journals.

Privacy-preserving Computational Geometry

Qiong Wei[1], Shundong Li[1], Wenli Wang[2], and Yanjing Yang[1]
(Corresponding author: Shundong Li)
School of Computer Science, Shaanxi Normal University, Xi'an 710119, China[1]
No.199, South Chang'an Road, Yanta District, Xi'an 710062
School of Mathematics and Information Science, Shaanxi Normal University, Xi'an 710119, China[2]
(Email: weiqiong@snnu.edu.cn)
(Received Nov. 20, 2018; Revised and Accepted May 18, 2019; First Online June 15, 2019)

Abstract

Secure multi-party computation (SMC) is a research hotspot in the field of international cryptography. Privacy-preserving computational geometry (PPCG) is the main branch of SMC. In this paper, we first design a protocol for deciding intersection of line segments, which can be used to determine whether polygons intersect. Then, we design a protocol to privately compute distance from a point to a plane, which is applicable to rational numbers or integers. We theoretically analyze the correctness, security and efficiency of the protocols. Based on the distance protocol, we construct a protocol to compute volume of a tetrahedron, and other two protocols to determine position relation between a line and a plane, and that between two planes. Finally, we analyze the efficiency of the protocol for determining position relation between a line and a plane and verify the analysis with experimental simulation.

Keywords: Cryptography; Distance from a Point to a Plane; Position Relation; Privacy-Preserving Computational Geometry; Secure Multi-party Computation

1 Introduction

SMC is a collaborative private computation between a group of non-trusted parties. It is an important technology of privacy protection in the information society and a research hotspot in the international cryptography field. SMC enables participants with private data to cooperate with each other in some joint computations without revealing their private data, thus enabling people to maximize the use of private data without compromising the privacy of the data. Therefore, SMC is widely used in data mining [20], data query [2], outsourcing computing [7] and so on [8,9]. SMC was first proposed by professor Yao [22], a Turing prize winner. Goldreich, Micali and others have done a lot of researches on the basis of Yao's work, which laid a theoretical foundation for SMC [19]. Goldwasser also predicted that SMC would become a vital part of computing science.

PPCG is an important area of SMC, which mainly studies the information security in the geometric cooperative computation. Du *et al.* [4] first introduced PPCG problem including point inclusion problem, intersection problem of two segments, intersection problem of two convex polygons, and convex hull problem of several secret points, and proposed solutions to these problems. Zhang *et al.* [23] studied the point inclusion problem. Liu *et al.* [12] studied the computation of triangle area in plane. Zuo *et al.* [24] solved the problem of whether three points are collinear based on Paillier homomorphic encryption scheme. Li *et al.* [10] proposed a secure solution to determine whether two graphics are similar. Luo *et al.* [13] studied the problem of determining the relationship between two lines, between a line and a plane, and between two planes. However, as the scheme calls for multiple basic protocols, such as comparing equal protocol, inner product protocol and data corresponding proportional protocol, the communication and computation costs of this scheme are very high. Li *et al.* [11] solved the determination of spatial position relation by computing the volume and height of tetrahedron. The scheme was efficient, but its ratio relation was disclosed when comparing the height of different tetrahedrons. Yang *et al.* [21] solved the decision problem of intersection between a straight line and a plane, in which the socialist millionaire protocol was invoked. The scheme requires multiple operations of encryption and decryption and cannot solve the rational number problem. Chen *et al.* [3] solved the problem of privately determining the position relations of various geometric objects in space. This scheme mainly uses the Boneh homomorphic encryption and inner product protocols, and outsources complex computing tasks to the cloud. However, the protocol brings additional high computational costs. If the computing is not outsourced to the cloud, the solution will be very inefficient. In addition, the scheme also converts the determination of the position relationship between a straight line and a plane and between two planes into an angle problem, thus revealing the angle information.

Sun *et al.* studied the problem of deciding intersec-

tion of line segments [18]. The scheme proposed by Sun *et al.* [18] invokes a complex millionaire protocol and discloses the endpoint information of these segments when looking for the intersection point. Luo *et al.* [14] and other existing schemes invoke many basic protocols, so the computational complexity is higher.

In this paper, we design a new protocol to determine whether two segments intersect, which solves the problems existing in previous schemes and can be used to determine the intersection of polygons. As for the secure computation problem of the distance from a point to a plane, the existing research uses inner product protocol in [13] to solve it, which has a high computational complexity. We design a secure computation protocol for distance from a point to a plane. On the basis of the distance problem, we design secure and efficient protocols to privately determine the position relation between a line and a plane, and between two planes in space.

Privately determining position relations of geometric objects is of great importance in practical applications. Consider the following two scenarios. Scenario 1: During the war, country A and country B are going to build a railway in country C, but the construction route will be kept secret until the railway is completed. In order to prevent future train collisions, countries A and B hope to determine whether the two routes will intersect without disclosing their own routes, so as to negotiate in advance and avoid accidents. Scenario 2: Two airlines have designed routes L_1 and L_2 between A and B. In order to ensure the safety of the routes, they need to determine whether the two routes will intersect, but in order not to lose the economic interests of the two airlines, they should not disclose their respective route information. Therefore, they want to determine whether L_1 and L_2 will intersect without disclosing their own route. In reality, many problems can be reduced to privately determine the position relations of geometric objects, so this problem has important research significance and value.

Our contributions: The main contributions of this paper are as follows.

1) In order to determine whether two segments intersect, we design a protocol based on the Paillier encryption algorithm, which avoids calling millionaire protocol and improves the efficiency;
2) In order to compute the distance from a point to a plane privately, we design an efficient protocol based on the Paillier homomorphic encryption algorithm, which solves the problem that the Paillier algorithm cannot directly encrypt non-integers, so that the protocol can solve not only the integer problem, but also the rational number problem;
3) By using the protocol for distance from a point to a plane, we further discuss and solve the problem of privately computing the volume of tetrahedron, the problem of privately determining position relation between a line and a plane, and between two planes;
4) In this paper, only a small amount of encryption and decryption operations are needed in the protocol to privately determine position relation between a line and a plane, and between two planes. In addition, the angle between line and plane will not be disclosed when the line intersects plane and the two planes intersect. Therefore, our protocol is secure and efficient.

Paper organization: The rest of this paper is organized as follows: Section 2 introduces some preliminaries. Section 3 presents a protocol for secure line segment intersection problem. Section 4 gives a protocol for secure distance from a point to a plane. Section 5 gives some applications of the protocol for the secure distance from a point to a plane. Section 6 analyses the efficiency of our protocols. Section 7 concludes the paper.

2 Preliminaries

2.1 Security

Semi-honest parties [6]. The protocols and securities proposed in this study are all based on a semi-honest model. A semi-honest party will follow the prescribed protocol exactly, but he may record the results of all intermediate computations and try to derive other parties' private inputs from the record. Goldreich has proved that, a protocol which can privately compute a function f in the semi-honest model can be complied, by introducing a bit commitment macro, into another protocol which can compute the function f in the malicious model. The semi-honest model is not only an important methodological tool but also provides a good model in many settings. It suffices to prove that a protocol is secure in the semi-honest model.

Two-party computation. Two-party computation represents a randomized computation process that maps a random input pair to an output pair: $f : \{0,1\}^* \times \{0,1\}^* \to \{0,1\}^* \times \{0,1\}^*$. This implies that given an input pair (x, y), the function will output two random variables $(f_1(x, y), f_2(x, y))$. The function is denoted by $f : (x, y) \to (f_1(x, y), f_2(x, y))$.

Privacy by simulation [17]. At present, simulations are used widely to prove the security of SMC protocols by simulating the execution process of SMC. The mathematical expression for the simulation is as follows.

Alice and Bob want to compute function f privately. Assume that $f = (f_1, f_2) : \{0,1\}^* \times \{0,1\}^* \to \{0,1\}^* \times$

$\{0,1\}^*$ is a probabilistic polynomial time function and that π represents a two-party protocol for computing f. When the input is (x,y), the message sequence obtained from the execution of protocol π is denoted by $view_i^{\pi}(x,y) = (x, r^i, m_1^i, m_2^i, \cdots, m_t^i)$, where i represents the ith participant, r^i represents the random number generated by the ith participant, and m_j^i represents the jth message that the ith participant obtains. The output of the participant is denoted by $output_i^{\pi}(x,y)(i=1,2)$.

Definition 1. *For a function $f(x,y)$, π privately computes f if probabilistic polynomial time algorithms S_1 and S_2 exist such that*

$$\{S_1(x, f_1(x,y))\}_{x,y} \stackrel{c}{\equiv} \{view_1^{\pi}(x,y)\}_{x,y} \quad (1)$$

$$\{S_2(y, f_2(x,y))\}_{x,y} \stackrel{c}{\equiv} \{view_2^{\pi}(x,y)\}_{x,y} \quad (2)$$

where $\stackrel{c}{\equiv}$ represents the computational indistinguishability.

2.2 Homomorphic Encryption

Homomorphic encryption [5] plays a crucial role in SMC and cloud computing security. One of the most important properties of homomorphic encryption is that it can perform some operations on the ciphertext without knowing the decryption key to ensure the privacy of the plaintext. Additively and multiplicatively homomorphic encryption are two general homomorphic types used in current researches. Paillier [16] designs an additively homomorphic encryption scheme that satisfies

$$\begin{aligned} E(m_1) \cdot E(m_2) &= E(m_1 + m_2) \\ (E(m_1))^{m_2} &= E(m_1 \cdot m_2). \end{aligned}$$

The Paillier cryptosystem can be constructed as follows.

Setup. Given a security parameter k, let $N = p \times q$, where p and q are two large primes, $\lambda = lcm(p-1, q-1)$ is the least common multiple of $p-1$ and $q-1$. Choose a $g \in Z_N^*$ at random such that $\gcd(L(g^{\lambda} \bmod N^2), N) = 1$, where $L(x) = \frac{x-1}{N}$. The public key of the cryptosystem is (g, N), and the private key is λ.

Encryption. To encrypt message $m < N$, choose a random number $r < N$, and compute

$$c = E(m) = g^m r^N \mod N^2$$

Decryption. Compute

$$m = D(c) = \frac{L(c^{\lambda} \bmod N^2)}{L(g^{\lambda} \bmod N^2)} \mod N^2$$

2.3 Distance from a Point to a Plane

Given a plane $Ax + By + Cz + D = 0$, the distance from point $P_0(x_0, y_0, z_0)$ to the plane can be denoted by:

$$d = \frac{|Ax_0 + By_0 + Cz_0 + D|}{\sqrt{A^2 + B^2 + C^2}} \quad (3)$$

2.4 Volume of Tetrahedron

Let the bottom area of the tetrahedron be S and the height be d. The volume of an arbitrary tetrahedron can be denoted by:

$$V = \frac{1}{3} Sd. \quad (4)$$

3 Privately Determine whether Two Segments Intersect

Suppose that Alice and Bob have segments L_1 and L_2, respectively. They want to know whether L_1 intersects with L_2 without disclosing any other information.

3.1 Basic Principle

Suppose that the two endpoints of L_1 and L_2 are $P_1(x_1, y_1), P_2(x_2, y_2)$ and $P_3(x_3, y_3), P_4(x_4, y_4)$, respectively, and the equations of the two straight line L_1 and L_2 are $l_1 : y = f_1(x) = k_1 x + b_1$, $l_2 : y = f_2(x) = k_2 x + b_2$, respectively. For L_1 and L_2, first determine whether L_1 intersects with straight line l_2, that is, whether P_1 and P_2 are on both sides of l_2. If they are on the same side, then the two segments do not intersect. Otherwise, continue to determine whether P_3 and P_4 are on both sides of l_1. If they are on both sides, then L_1 intersects with L_2, otherwise it does not. Therefore, the problem is converted into determining whether a segment intersects with a straight line. If $(y_1 - f_2(x_1)) \times (y_2 - f_2(x_2)) \le 0$, then P_1 and P_2 are located on both sides of the straight line l_2 (including a point on the line). (When the slope of the straight line does not exist, the line is a vertical line of $x = C$, where C is a constant, just determine whether $(x_1 - x) \times (x_2 - x) \le 0$. This paper only considers general situations.) Therefore, we compute

$$\begin{aligned} m &= (y_1 - f_2(x_1)) \times (y_2 - f_2(x_2)) \\ &= u_1 + U_1 + u_2 U_2 + u_3 U_3 + u_4 U_4 + u_5 U_5 \end{aligned}$$

where $u_1 = y_1 y_2, u_2 = x_2 y_1 + x_1 y_2, u_3 = y_1 + y_2, u_4 = x_1 x_2, u_5 = x_1 + x_2, U_1 = {b_2}^2, U_2 = -k_2, U_3 = -b_2, U_4 = {k_2}^2, U_5 = k_2 b_2$. If $m \le 0$, then P_1, P_2 are on both sides of l_2.

Similarly, if $(y_3 - f_1(x_3)) \times (y_4 - f_1(x_4)) \le 0$, then P_3 and P_4 are located on both sides of l_1. Therefore, we compute

$$\begin{aligned} n &= (y_3 - f_1(x_3)) \times (y_4 - f_1(x_4)) \\ &= v_1 + V_1 + v_2 V_2 + v_3 V_3 + v_4 V_4 + v_5 V_5 \end{aligned}$$

where $v_1 = y_3 y_4, v_2 = -(x_3 y_4 + x_4 y_3), v_3 = -(y_3 + y_4), v_4 = x_3 x_4, v_5 = x_3 + x_4, V_1 = {b_1}^2, V_2 = k_1, V_3 = b_1, V_4 = {k_1}^2, V_5 = k_1 b_1$. If $n \le 0$, then the two endpoints P_3, P_4 are on both sides of l_1.

L_1 intersects with L_2 if and only if $m \le 0$ and $n \le 0$.

In addition, an important step in determining whether two polygons intersect is to determine whether they have

a set of intersecting edges. Therefore, the method to determine whether two segments intersect can also be used to determine whether two polygons intersect.

For simple exposition, we define

$$P(L_1, L_2) = \begin{cases} 0, & L_1 \text{ intersects with } L_2 \\ 1, & \text{otherwise} \end{cases}$$

Proposition 1. *For the Paillier cryptosystem, $N = p \times q$, where p and q are two large primes. Suppose that $0 \leq u, v < N/2$, $C = E(u)E(N - v)$ and $w = D(C)$, we have the following conclusions: $u = v$ if and only if $w = 0$; $u > v$ if and only if $0 < w < N/2$; $u < v$ if and only if $w > N/2$.*

Proof. According to the definition and additive homomorphism of the Paillier encryption algorithm we know that $C = E(u)E(N - v) = E(N + u - v \bmod N)$. Then we decrypt C with private key and get $w = D(C) = (u - v) \bmod N$.

1) when $u = v$, $w = D(C) = (u - v) \bmod N = 0$.
2) when $u > v$, because $0 < u - v < N/2$, $w = (u - v) \bmod N = u - v < N/2$.
3) when $u < v$, because $-N/2 < u - v < 0$, $w = (u - v) \bmod N = N + u - v > N/2$.

Because $w = D(C) = (u-v) \bmod N \in Z_N$ and $0 \leq u, v < N/2$, w can only get one of the three results: $w = 0, 0 < w < N/2$ or $N/2 < w < N$. This completes the proof of the proposition. □

3.2 Protocol Design

Protocol 1: Privately determine whether two segments intersect.

Inputs: Private segments L_1 and L_2.

Output: $P(L_1, L_2)$.

1) Alice generates the public key and private key of the Paillier encryption scheme, and tells the public key to Bob.
2) Alice takes the two endpoints $P_1(x_1, y_1), P_2(x_2, y_2)$ of segment L_1, and computes $u_1 = y_1y_2, u_2 = x_2y_1 + x_1y_2, u_3 = y_1 + y_2, u_4 = x_1x_2, u_5 = x_1 + x_2$. Then, Alice encrypts u_1, u_2, u_3, u_4, u_5 with public key to get $E(u_1), E(u_2), E(u_3), E(u_4), E(u_5)$, and sends the ciphertexts to Bob.
3) Bob first computes $U_1 = {b_2}^2$, $U_2 = -k_2$, $U_3 = -b_2$, $U_4 = {k_2}^2$, $U_5 = k_2b_2$, and then computes $Z_1 = E(u_1)E(U_1)E(u_2)^{U_2}E(u_3)^{U_3}E(u_4)^{U_4}E(u_5)^{U_5}E(N)$. Bob sends Z_1 to Alice.
4) Alice decrypts Z_1 to get z_1.
5) If $z_1 \in (0, N/2)$, then segments L_1 and L_2 do not intersect. Alice outputs $P(L_1, L_2) = 1$. Otherwise, they proceed with the following steps.
6) Alice computes $V_1 = {b_1}^2$, $V_2 = k_1$, $V_3 = b_1$, $V_4 = {k_1}^2$, $V_5 = k_1b_1$, and encrypts V_1, V_2, V_3, V_4, V_5 to get $E(V_1), E(V_2), E(V_3), E(V_4), E(V_5)$. Alice sends ciphertexts to Bob.
7) Bob first computes $v_1 = y_3y_4$, $v_2 = -(x_3y_4 + x_4y_3)$, $v_3 = -(y_3 + y_4)$, $v_4 = x_3x_4$, $v_5 = x_3 + x_4$, and then computes $Z_2 = E(V_1)E(v_1)E(V_2)^{v_2}E(V_3)^{v_3}E(V_4)^{v_4}E(V_5)^{v_5} \cdot E(N)$. Bob sends Z_2 to Alice.
8) Alice decrypts Z_2 to get z_2.
9) If $z_2 \in (0, N/2)$, then L_1 and L_2 do not intersect. Alice outputs $P(L_1, L_2) = 1$. Otherwise, Alice outputs $P(L_1, L_2) = 0$.

3.3 Correctness

Theorem 1. *Protocol 1 can correctly determine whether two segments intersect.*

Proof. According to the additive homomorphism of the Paillier encryption algorithm, we can get $Z_1 = E(u_1)E(U_1)E(u_2)^{U_2}E(u_3)^{U_3}E(u_4)^{U_4}E(u_5)^{U_5}E(N) = E(u_1U_1 + u_2U_2 + u_3U_3 + u_4U_4 + u_5U_5 + N) = E(m + N)$. Alice decrypts Z_1 to get z_1. According to Proposition 1 we can see that: if $m > 0$, then $(m + N) \bmod N < N/2$; if $m = 0$, then $(m + N) \bmod N = 0$; otherwise, $(m + N) \bmod N > N/2$. Therefore, Alice can determine whether m is positive or negative based on z_1 according to Proposition 1. According to the calculation principle, we can correctly determine whether the two endpoints P_1 and P_2 of L_1 are on both sides of the straight line l_2 by judging whether m is positive or negative. Similarly, Alice can determine whether the two endpoints P_3 and P_4 of L_2 are on both sides of the straight line l_1 by judging whether n is positive or negative. Therefore, Protocol 1 can correctly determine whether two segments intersect. This completes the proof of the theorem. □

3.4 Security

By the simulation paradigm, we can prove that Protocol 1 is secure. The theorem is proved by constructing simulators S_1 and S_2 such that Equations (1) and (2) hold.

Theorem 2. *Protocol 1 for determining whether two segments intersect is secure.*

Proof.

1) S_1 selects an arbitrary segment L_2' (with coordinates of $P_3'(x_3', y_3'), P_4'(x_4', y_4')$ and linear equation $y' = f_2'(x) = k_2'x + b_2'$) such that $P(L_1, L_2') = P(L_1, L_2)$.
2) S_1 computes the linear equation $y' = f_2'(x) = k_2'x + b_2'$ of L_2' according to P_3', P_4'.
3) S_1 computes $U_1' = {b_2'}^2$, $U_2' = -k_2'$, $U_3' = -b_2'$, $U_4' = {b_2'}^2$, $U_5' = k_2'b_2'$ and computes $Z_1' = E(u_1)E(U_1')E(u_2)^{U_2'}E(u_3)^{U_3'}E(u_4)^{U_4'}E(u_5)^{U_5'}E(N)$.

4) S_1 decrypts Z_1' to obtain z_1'. If the protocol is terminated at this time,

$$view_1^{\pi}(L_1, L_2) = \{L_1, Z_1, P(L_1, L_2)\}.$$

The information sequence generated in the simulation process is: $S_1(L_1, f_1(L_1, L_2)) = \{L_1, Z_1', P(L_1, L_2')\}$.

According to the computation process, for Alice: $Z_1' \stackrel{c}{\equiv} Z_1$, and $P(L_1, L_2') = P(L_1, L_2)$, therefore

$$\{S_1(L_1, f_1(L_1, L_2))\} \stackrel{c}{\equiv} \{view_1^{\pi}(L_1, L_2)\}.$$

Simulator S_2 can be constructed like this, and the following formula holds

$$\{S_2(L_2, f_2(L_1, L_2))\} \stackrel{c}{\equiv} \{view_2^{\pi}(L_1, L_2)\}.$$

If it is impossible to determine whether the two segments intersect at this point, then the second part of the protocol needs to be performed. The following simulation process is as follows:

5) S_1 computes $v_1' = y_3'y_4'$, $v_2' = -(x_3'y_4' + x_4'y_3')$, $v_3' = -(y_3'+y_4')$, $v_4' = x_3'x_4'$, $v_5' = x_3'+x_4'$, and computes $Z_2' = E(V_1)E(v_1')E(V_2)^{v_2'}E(V_3)^{v_3'}E(V_4)^{v_4'}E(V_5)^{v_5'}E(N)$.

6) S_1 decrypts Z_2' to get z_2'.

During the execution of Protocol 1,

$$view_1^{\pi}(L_1, L_2) = \{L_1, Z_1, Z_2, P(L_1, L_2)\}.$$

The information sequence generated in the simulation process is:

$$S_1(L_1, f_1(L_1, L_2)) = \{L_1, Z_1', Z_2', P(L_1, L_2')\}.$$

From the above part of proof, we can see: $Z_1' \stackrel{c}{\equiv} Z_1$; according to the computation process, for Alice: $Z_2' \stackrel{c}{\equiv} Z_2$ and $P(L_1, L_2') = P(L_1, L_2)$, therefore

$$\{S_1(L_1, f_1(L_1, L_2))\} \stackrel{c}{\equiv} \{view_1^{\pi}(L_1, L_2)\}.$$

Similarly, simulator S_2 can be constructed like this, and the following formula holds

$$\{S_2(L_2, f_2(L_1, L_2))\} \stackrel{c}{\equiv} \{view_2^{\pi}(L_1, L_2)\}.$$

This completes the proof of the theorem. □

4 Privately Compute the Distance from a Point to a Plane

Suppose that Alice has a plane π: $Ax + By + Cz + D = 0$ and Bob has a point $P_0(x_0, y_0, z_0)$. They want to know the distance between P_0 and π without disclosing other information about the point and the plane.

4.1 Basic Principle

According to Equation (3), the distance from a point to a plane can be computed directly, but there is no secrecy in doing so. Therefore, we design a secure Protocol 2 to compute the distance. Protocol 2 is mainly implemented by using the Paillier homomorphic encryption algorithm. Suppose that A, B, C, and D in Equation (3) are all rational numbers, since the Paillier encryption algorithm cannot directly encrypt rational numbers, we can transform them into integers for processing. Thus, we multiply A, B, C, and D by the least common multiple m of their denominator (if A, B, C, D are all integers, $m = 1$), and do the same for x_0, y_0, z_0. (If x_0, y_0, z_0 are integers, the least common multiple of denominator $n = 1$).

4.2 Protocol Design

Protocol 2: Privately compute distance from a point to a plane.

Inputs: Private plane π: $Ax + By + Cz + D = 0$ and point $P_0(x_0, y_0, z_0)$.

Output: The distance d from P_0 to π.

1) Alice generates the public key and private key of the Paillier homomorphic encryption scheme, and tells the public key to Bob.

2) Alice computes the least common multiple m of four rational denominators A, B, C, and D (when A, B, C, D are integers, $m = 1$), then computes $A' = A \cdot m$, $B' = B \cdot m$, $C' = C \cdot m$, $D' = D \cdot m$. Alice encrypts plane π with public key to obtain $E(\pi) = (E(A'), E(B'), E(C'))$, and sends $E(\pi)$ to Bob.

3) Bob finds out the least common multiple n of three rational denominators (when x_0, y_0, z_0 are integers, $n = 1$), and chooses random numbers r_1, r_2 to compute $x_0' = x_0 \cdot r_1 n$, $y_0' = y_0 \cdot r_1 n$, $z_0' = z_0 \cdot r_1 n$, then Bob computes $T = E(A')^{x_0'} \cdot E(B')^{y_0'} \cdot E(C')^{z_0'} \cdot {r_2}^N \bmod N^2$. Bob sends T and $r_1 n$ to Alice.

4) Alice decrypts T with private key and gets $T' = D(T) = A'x_0' + B'y_0' + C'z_0'$. Then Alice computes $d = \frac{|T' + r_1 n D'|}{m r_1 n \cdot \sqrt{A^2+B^2+C^2}}$, and tells Bob the result.

4.3 Correctness

Theorem 3. *Protocol 2 can correctly get the distance from a point to a plane.*

Proof. Since each rational number can be expressed as a fraction, Alice turns the rational number into an integer by multiplying A, B, C, D by the least common multiple m of their denominator. Similarly, Bob multiplies x_0, y_0, z_0 by the least common multiple n of their denominators. According to the homomorphism of the Paillier

encryption algorithm,

$$\begin{aligned} T &= E(A')^{x_0'} \cdot E(B')^{y_0'} \cdot E(C')^{z_0'} \cdot {r_2}^N \bmod N^2 \\ &= E(A'x_0' + B'y_0' + C'z_0') \end{aligned}$$

Alice decrypts the value of $A'x_0' + B'y_0' + C'z_0'$, then computes:

$$\begin{aligned} d &= \frac{|T' + r_1 n D'|}{mr_1 n \cdot \sqrt{A^2 + B^2 + C^2}} \\ &= \frac{|Amr_1 nx_0 + Bmr_1 ny_0 + Cmr_1 nz_0 + Dmr_1 n|}{mr_1 n \cdot \sqrt{A^2 + B^2 + C^2}} \\ &= \frac{|Ax_0 + By_0 + Cz_0 + D|}{\sqrt{A^2 + B^2 + C^2}} \end{aligned}$$

It can be seen from the above formula that m, n do not affect the final result. This completes the proof of the theorem. □

4.4 Security

The security of Protocol 2 is based on the security of the Paillier homomorphic encryption algorithm, which has semantic security. By the simulation paradigm, we can prove that Protocol 2 is secure.

Theorem 4. *Protocol 2 for computing the distance from a point to a plane is secure.*

Proof. The theorem is proved by constructing simulators S_1 and S_2 that make Equations (1) and (2) hold. □

1) S_1 accepts input $(\pi, f_1(\pi, P_0))$, and selects a point $P_1(x_1, y_1, z_1)$ such that $f_1(\pi, P_1) = f_1(\pi, P_0)$.

2) S_1 computes the least common multiple n' of the denominator of rational numbers x_1, y_1, z_1 and chooses a random number r_1' to compute $x_1' = x_1 r_1' n', y_1' = y_1 r_1' n', z_1' = z_1 r_1' n'$. Then S_1 chooses a random number r_2' to compute $T_1 = E(A')^{x_1'} \cdot E(B')^{y_1'} \cdot E(C')^{z_1'} {r_2'}^N \bmod N^2$. S_1 encrypts T_1 to get T_1', and finally computes $d' = \frac{|T_1' + r_1' n' D_1'|}{mr_1' n' \cdot \sqrt{A^2+B^2+C^2}}$.

$$view_1^{\pi}(\pi, P_0) = \{\pi, r_1 n, T, d\}$$

The information sequence generated in the simulation process is: $S_1(\pi, f_1(\pi, P_0)) = \{\pi, r_1' n', T_1', d'\}$,

By definition and the semantic security of the homomorphic encryption scheme, $f_1(\pi, P_1) = f_1(\pi, P_0)$, $T_1 \overset{c}{\equiv} T$, $d' = d$. Therefore,

$$\{S_1(\pi, f_1(\pi, P_0))\} \overset{c}{\equiv} \{view_1^{\pi}(\pi, P_0)\}$$

Similarly, we can construct S_2 such that

$$\{S_2(P_0, f_2(\pi, P_0))\} \overset{c}{\equiv} \{view_2^{\pi}(\pi, P_0)\}$$

This completes the proof of the theorem.

5 Application

Now, we can use the distance protocol to privately compute the volume of tetrahedron and determine the position relation between a line and a plane and between two planes.

5.1 Privately Compute the Volume of Tetrahedron

Suppose that Alice has several points in a plane π: $Ax + By + Cz + D = 0$, and Bob has a point $P_0(x_0, y_0, z_0)$(p_0 is not on the plane π). These points and P_0 constitute a tetrahedron. Alice and Bob want to compute the volume of the tetrahedron without disclosing any information about P_0 and π. The key to solve the problem is to get the height of tetrahedron, that is, the distance from point P_0 to plane π. Finally, the volume of tetrahedron can be calculated according to Equation (4).

Protocol 3: Privately compute the volume of tetrahedron.

Inputs: Private plane π: $Ax + By + Cz + D = 0$, private point $P_0(x_0, y_0, z_0)$.

Output: The volume V of tetrahedron formed by point P_0 and other points in π.

1) Alice and Bob invoke Protocol 1 to compute the distance d from $P_0(x_0, y_0, z_0)$ to π.

2) Alice obtains d and computes the area S of the bottom surface, then computes $V = \frac{1}{3}Sd$. Alice sends the result to Bob.

5.2 Privately Determine the Position Relations between a Straight Line and a Plane

Suppose that Alice has a plane π: $Ax + By + Cz + D = 0$ and Bob has a straight line L. They want to know the position relationship between L and π without disclosing any information about π and L.

5.2.1 Basic Principle

Choosing two different points $P_1(x_1, y_1, z_1), P_2(x_2, y_2, z_2)$ on the line L and comparing distances d_1 and d_2 from these two points to the plane. If $d_1 \neq d_2$, then line L intersects with plane π. If $d_1 = d_2 = 0$, then L is in π. If $d_1 = d_2 \neq 0$, then L is parallel to π. It is also important to note that when a straight line intersects a plane, the specific value of the distance cannot be computed directly, because this will disclose the angle between the line and the plane. Therefore, the relationship of the distance between different points to the plane should be kept secret in the protocol.

For simple exposition, we define

$$P(L,\pi)=\begin{cases}0, & L \text{ is in the } \pi\\ 1, & L \text{ is parallel to } \pi\\ 2, & L \text{ intersects with } \pi\end{cases}$$

5.2.2 Protocol Design

Protocol 4: Privately determine the position relation between a line and a plane.

Inputs: Private plane π: $Ax+By+Cz+D=0$ and straight line L.

Output: $P(L,\pi)$.

1) Alice generates the public key and private key of the Paillier homomorphic encryption scheme, and tells the public key to Bob.

2) Alice encrypts plane π with public key to obtain $E(\pi)=(E(A),E(B),E(C))$, and sends $E(\pi)$ to Bob.

3) Bob chooses two points $P_1(x_1,y_1,z_1), P_2(x_2,y_2,z_2)$ on the line L, then computes $t_1=E(A)^{x_1}\cdot E(B)^{y_1}\cdot E(C)^{z_1}$, $t_2=E(A)^{x_2}\cdot E(B)^{y_2}\cdot E(C)^{z_2}$, and computes $T_1=t_1\cdot t_2^{-1}$. Bob sends T_1 to Alice.

4) Alice decrypts T_1 to obtain $T_1'=D(T_1)=Ax_1+By_1+Cz_1-Ax_2-By_2-Cz_2$. If $T_1'\neq 0$, then $d_1\neq d_2$, L intersects π, Alice outputs $P(L,\pi)=2$. The protocol terminates. Otherwise, they continue to perform the next step.

5) Bob sends t_1 to Alice.

6) Alice decrypts t_1 to obtain $t_1'=Ax_1+By_1+Cz_1$, then computes $d_1=d_2=\frac{t_1'+D}{\sqrt{A^2+B^2+C^2}}$. If $d_1=d_2=0$, L is in the π. Alice outputs $P(L,\pi)=0$. If $d_1=d_2\neq 0$, then L is parallel to π. Alice outputs $P(L,\pi)=1$.

5.3 Privately Determine the Position Relation between Two Planes

Suppose that Alice has a plane π_1: $A_1x+B_1y+C_1z+D_1=0$ and Bob has a plane π_2: $A_2x+B_2y+C_2z+D_2=0$. They want to know the position relation between π_1 and π_2 without disclosing any information about the two planes.

5.3.1 Basic Principle

Since two intersecting lines can determine a plane, we choose two intersecting lines L_1 and L_2 in the plane π_2, and then the problem of determining the position relation between two planes is transformed into the problem of determining the position relation between a line and a plane. Therefore, we can call protocol 4 to solve this problem. For simplicity, select the intersection point P_0 of two intersecting lines as one of the points. In addition, select the other point in two straight lines. If the distances d_0,d_1 from two points P_0,P_1 on the line L_1 to the plane π_1 and the distances d_0,d_2 from points P_0,P_2 on the line L_2 to the plane π_1 satisfy that $d_0=d_1=d_2=0$, then π_1 coincides with π_2. If $d_0=d_1=d_2\neq 0$, then π_1 is parallel to π_2. Otherwise, π_1 intersects with π_2.

For simple exposition, we define:

$$P(\pi_1,\pi_2)=\begin{cases}0, & \pi_1 \text{ and } \pi_2 \text{ coincide}\\ 1, & \pi_1 \text{ is parallel to } \pi_2\\ 2, & \pi_1 \text{ intersects with } \pi_2\end{cases}$$

5.3.2 Protocol Design

Protocol 5: Privately determine the position relation between two planes.

Inputs: Private plane π_1: $A_1x+B_1y+C_1z+D_1=0$ and π_2: $A_2x+B_2y+C_2z+D_2=0$.

Output: $P(\pi_1,\pi_2)$.

1) Bob chooses two intersecting lines L_1 and L_2 in the plane π_2.

2) Alice and Bob invoke Protocol 4 to compare the distances d_0,d_1 from two points P_0,P_1 on the line L_1 to the plane π_1 and the distances d_0,d_2 from two points P_0,P_2 on the line L_2 to the plane π_1.

3) If $d_0=d_1=d_2=0$, then π_1 coincides with π_2, Alice outputs $P(\pi_1,\pi_2)=0$. If $d_0=d_1=d_2\neq 0$, then π_1 is parallel to π_2, Alice outputs $P(\pi_1,\pi_2)=1$. Otherwise, π_1 intersects with π_2, Alice outputs $P(\pi_1,\pi_2)=2$.

6 Performance Analysis

6.1 Efficiency Analysis

Computational complexity analysis. At present, the solutions to privately determine whether two segments intersect need to invoke the complex millionaire protocol, oblivious transfer, inner product protocol, etc. One of the most efficient schemes is the protocol in [18], which uses the Paillier encryption algorithm and needs to be encrypted twice and decrypted 4 times. Encryption or decryption using the Paillier encryption algorithm requires 2 modular exponentiations at a time. In addition, the millionaire protocol based on the Paillier homomorphic encryption scheme [1] is invoked 3 times, and each invocation requires $4n$ modular exponentiations (n is the bit length of the input data). Ignoring multiplication and addition operations, the total computational overheads are $4n+12$ modular exponentiations. In this paper, Protocol 1 needs to be encrypted 12 times and decrypted twice, so the total computational overheads are 28 modular exponentiations. From the analysis above, we can see that with the growth of data, the efficiency of our scheme has obvious advantages. (The modular exponentiation is M_e.)

For the research of privacy-preserving computation of the distance from a point to a plane, the scheme in [13] needs to invoke the inner product protocol based on oblivious transfer. Assuming that the security parameter is m. To execute an inner product protocol needs to invoke the 1-out-of-k oblivious transfer m times, i.e. $\lg k$ 1-out-of-k oblivious transfer, i.e. $2m \lg k$ modular exponentiations. According to the practical significance, only when $m > 5$ and $k > 8$ can the scheme achieve the basic security level. Thus, the scheme requires at least 30 modular exponentiations. In this paper, we design Protocol 2 for computing the distance from a point to a plane. Protocol 2 needs to be encrypted 3 times and decrypted once, so the total computational overheads are 8 modular exponentiations.

Communication complexity analysis. The measure of communication complexity is the number of bits of information exchanged in the protocol or the number of communication rounds. In the study of SMC, the number of communication rounds is usually used. The scheme in [18] requires $4 + 3c$ rounds of communication where c represents the number of rounds of millionaire protocol. The communication complexity of Protocol 1 in this paper is 2 rounds. [13] calls m times inner product protocol based on oblivious transfer. The communication complexity is m rounds. The communication complexity of Protocol 2 is 2 rounds. The computational and communication complexity are shown in Table 1.

In addition, we give an analysis of the efficiency and security of Protocol 4 in the application part and compare it with the related protocols in [3, 11, 13, 21]. [13] and [3] invoke the inner product protocol, and [13] and [21] invoke the data ratio protocol. In the whole process of execution, the most expensive computation cost is modular exponentiation. [11] mainly uses multiplication. The total number of inner product protocol called in each scheme, the number of modular exponentiations required by the user, and the number of multiplication operations are taken as indicators to measure the complexity of computation, and the others are ignored. The modular exponentiation is M_e and multiplication is M.

Privately determining the position relations between a line and a plane in [13] (Protocol 6): The protocol invokes the inner product protocol twice and the data ratio protocol twice, and the inner product protocol uses the oblivious transfer method in [15]. Assuming the security parameter is m, according to the analysis in the original paper, it needs at least $30m$ modular exponentiations, so the computational cost of the protocol is $30mM_e$. [11] (Protocol 3): This protocol mainly performs matrix operations, and the total number of multiplication operations is $36M$. However, the protocol will disclose the ratio between the distance from different points on the line to the plane. [21] (Protocol 4): This scheme mainly uses the Paillier homomorphic encryption algorithm, and calls data ratios protocol twice. The total computational overheads are $35M_e$. [3] (Protocol 5): The scheme invokes inner product protocol twice and outsources them to cloud computing, and invokes data ratio protocol once. The total modular exponentiations are $15M_e$. However, when the line and plane are intersected, this scheme will disclose the angle between the line and the plane. In this paper (Protocol 4), the total number of modular exponentiations is $16M_e$ and the total number of multiplication operations is $4M$.

As for the problem of privately determining position relation between a line and a plane, the comparisons are shown in Table 2.

6.2 Experimental Test and Analysis

We verify the computational complexity by simulating the time taken to perform Protocol 2, and compare it with the existing scheme in [13]. In addition, we test the time used in Protocol 4 to determine the position relation between a line and a plane. [3] shifts this decision problem to cloud computing platforms and outsources the complex computation to the cloud, which results in additional high economic costs. What's more, in the case of traditional participant interaction, the computational complexity of this scheme is still very high, and it is found that both the schemes in [11] and [3] have information leakage. Therefore, we choose [21] which is the most efficient solution and without information leakage to compare with Protocol 4 in the mode of participant interaction.

Our test environment: Windows 10 64 bit operating system. The processor is Intel (R) Core (TM) i5-6600 CPU @3.30 HZ, and memory is 8GB. We program in JAVA language.

Experimental method. We randomly selected 20 sets of data, conducted 2000 simulation experiments on each set value, and calculated the average of experimental results. Figure 1 depicts the comparison of the execution time between Protocol 2 and the scheme in [13]. Figure 2 describes the comparison of the implementation time between Protocol 4 and the scheme in [21].

The experimental results show that the average execution time of Protocol 2 is between 15 and 25 milliseconds, which is much more efficient than the method in [13]. At the same time, Protocol 4 can guarantee the security with high efficiency. To sum up, the computation cost and computational complexity of our protocols are relatively low.

7 Conclusion

PPCG has always been an important issue in cryptography and SMC. Based on the Paillier homomorphic encryption scheme, we first proposed a secure Protocol 1 to determine whether two segments intersect. By using the

Table 1: Comparison of computational and communication complexity between our protocols and existing schemes

	[18]	Protocol 1	[13]	Protocol 2
Computational Complexity	$(4n+12)M_e$	$28M_e$	$2m \lg k > 30M_e$	$8M_e$
Communication Complexity	2 rounds	$4+3c$ rounds	m rounds	2 rounds

Table 2: Efficiency and performance comparison between Protocol 4 and existing protocols

	Computation overhead	Inner product protocol	Data ratio protocol	Information leakage?
Protocol 6 in [13]	$30mM_e$	twice	twice	No
Protocol 3 in [11]	$36M$	-	-	Yes
Protocol 4 in [21]	$35M_e$	-	twice	No
Protocol 5 in [3]	$15M_e$	twice (cloud computing)	once	Yes
Protocol 4	$16M_e+4M$	-	-	No

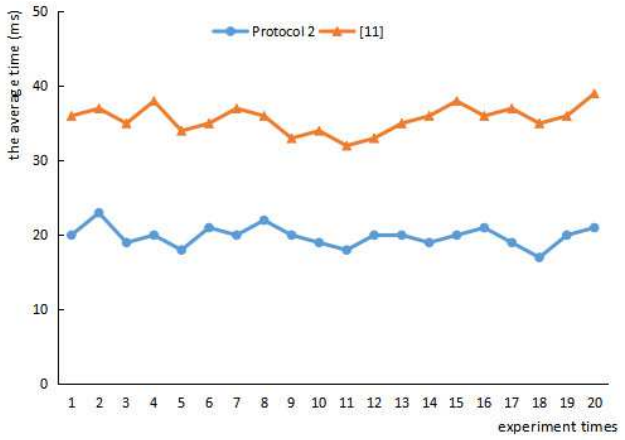


Figure 1: Comparison of execution time between Protocol 2 and [13]

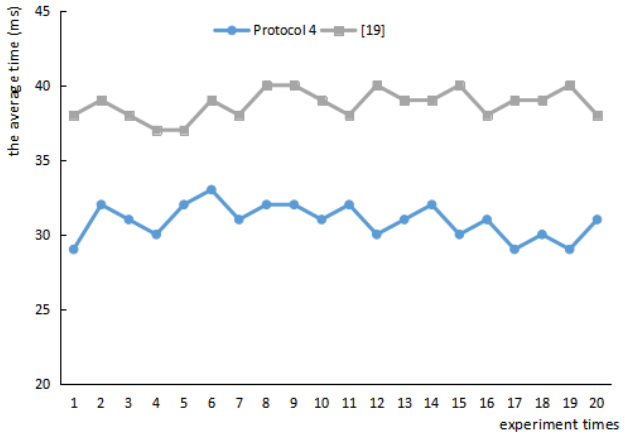


Figure 2: Comparison of execution time between Protocol 4 and [21]

principle of Protocol 1, we can determine whether polygons intersect. Then, we propose a secure Protocol 2 to compute the distance from a point to a plane. Protocol 2 is not only suitable for integers, but also for rational numbers. The correctness of Protocol 1 and Protocol 2 are analyzed and proved, and the security of the two protocols is proved by simulation paradigm. Next, by using the distance from a point to a plane, we solve the problem of privately computing volume of a tetrahedron, the problem of privately determining position relation between a line and a plane, and the problem of privately determining position relation between two planes in space. Finally, we prove that Protocol 4 is not only efficient but also secure by comparing with the existing protocols for privately determining the position relation between a line and a plane in space. The problems studied in this paper have important practical significance for research and application of SMC, and our solutions for these problems are secure in the semi-honest model. In the future study, we will focus on the SMC of various computational geometry problems in the malicious models.

Acknowledgments

The authors would like to thank the anonymous reviewers for detailed and valuable comments. This work is supported by the National Natural Science Foundation of China (Grant no. 61272435).

References

[1] I. F. Blake, V. Kolesnikov, "Strong conditional oblivious transfer and computing on intervals," in *Advances in Cryptology-Asiacrypt, International Conference on the Theory and Application of Cryptology and Information Security*, pp. 515–529, 2004.

[2] R. Campos and A. Jatowt, "Survey of temporal information retrieval and related applications," *Acm Computing Surveys*, vol. 47, no. 2, pp. 1–41, 2014.

[3] Z. H. Chen, S. D. Li, Q. Huang, Y. Ding and M. Sun, "Privacy-preserving determination of spatial location-relation in cloud computing," *Chinese Journal of Computers*, vol. 39, no. 137, pp. 351–363, 2017.

[4] W. Du and M. J. Atallah, "Secure multi-party computation problems and their applications: A review and open problems," *Proceedings of New Security Paradigms workshop New York: ACM Press*, pp. 13–22, 2001. ISBN:1-58113-457-6.

[5] R. Frederick, "Core concept: Homomorphic encryption," *Proceedings of the National Academy of Science*, vol. 112, no. 28, pp. 8515–8516, 2015.

[6] O. Goldreich, "Foundations of cryptography: Casic applications," *Journal of the Acm*, vol. 10, no. 509, pp. 359–364, 2004.

[7] F. Kerschbaum, "Privacy-preserving computation," *Springer Berlin Heidelberg*, vol. 8319, pp. 41–54, 2012.

[8] C. T. Li, M. S. Hwang, Y. P. Chu, "A secure and efficient communication scheme with authenticated key establishment and privacy preserving for vehicular ad hoc networks", *Computer Communications*, vol. 31, no. 12, pp. 2803-2814, July 2008.

[9] C. T. Li, M. S. Hwang, Y. P. Chu, "Further improvement on a novel privacy preserving authentication and access control scheme for pervasive computing environments", *Computer Communications*, vol. 31, no. 18, pp. 4255–4258, Dec. 2008.

[10] S. D. Li, X. L. Yang, X. J. Zuo, *et al.*, "Privacy protecting similitude determination for Graphics Similarity," *Chinese Journal of Electronics*, vol. 45, no. 9, pp. 2184–2189, 2017.

[11] S. D. Li, C. Y. Wu, D. S. Wang, and Y. Q. Dai, "Secure multiparty computation of solid geometric problems and their applications," *Information Sciences An International Journal*, vol. 282, pp. 401–413, 2014.

[12] L. Liu, X. Chen, and W. Lou, "Secure three-party computational protocols for triangle area," *International Journal of Information Security*, vol. 15, no. 1, pp. 1–13, 2016.

[13] Y. L. Luo, L. S. Huang, W. W. Jing, and W. J. Xu, "Privacy protection in the relative position determination for two spatial geometric," *Journal of Computer Research and Development*, vol. 43, no. 3, pp. 410–416, 2006.

[14] Y. L. Luo, L. S. Huang, W. W. Jing, W. J. Xu, and G. L. Chen, "Privacy-preserving cross product protocol and its applications," *Chinese Journal of Computers*, vol. 30, no. 2, pp. 248–254, 2007.

[15] M. Naor, B. Pinkas, "Oblivious transfer and polynomial evaluation," *ACM Symposium on Theory of Computing*, pp. 245–254, 1999. ISBN:1-58113-067-8.

[16] P. Paillier, "Public-key cryptosystems based on composite degree residuosity classes," in *International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 223–238, 1999.

[17] B. Reimer, R. Fried, B. Mehler, *et al.*, "Brief report: Examining driving behavior in young adults with high functioning autism spectrum disorders: A pilot study using a driving simulation paradigm," *Journal of Autism and Developmental Disorders*, vol. 43, no. 9, pp. 2211–2217, 2013.

[18] M. H. Sun, S. S. Luo, Y. Xin, and Y. X. Yang, "Secure two-party line segments intersection scheme and its application in privacy-preserving convex hull intersection," *Journal on Communications*, vol. 34, no. 1, pp. 30–42, 2013.

[19] M. Vishakha and R. Sharma, "Review paper on cryptography," *International Journal of Research*, vol. 2, no. 5, pp. 141–142, 2015.

[20] J. Yang, J. S. Zhao and J. P. Zhang, "A privacy preservation method for high dimensional data mining," *Acta Electronica Sinica*, vol. 41, no. 11, pp. 2187–2192, 2008.

[21] X. L. Yang, S. D. Li and X. J. Zuo, "Secure multiparty geometry computation," *Journal of Cryptologic Research*, vol. 3, no. 1, pp. 33–41, 2016.

[22] A. C. Yao, "Protocols for secure computations," in *IEEE Computer Science*, pp. 160–164, 1982.

[23] J. Zhang, S. S. Luo, Y. X. Yang, and Y. Xin, "Research on the privacy-preserving point-in-polygon protocol," *Journal on communications*, vol. 37, no. 4, pp. 87–95, 2016.

[24] X. J. Zuo, X. L. Yang, and S. D. Li, "Privately determining protocol on three points are collinear and its applications," *Journal of Cryptologic Research*, vol. 3, no. 3, pp. 238–248, 2016.

Biography

Qiong Wei was born in 1994. She is currently pursuing the M.S. degree with School of Computer Science in Shaanxi Normal University. Her research interests focus on secure multi-party computation and information security.

Shundong Li was born in 1963. He received the Ph.D. degree in Department of Computer Science and Technology from Xi'an Jiaotong University in 2003. He is now a Professor with School of Computer Science in Shaanxi Normal University. His research interests focus on modern cryptography and secure multi-party computation.

Wenli Wang was born in 1991. She is currently pursuing the M.S. degree with School of Mathematics and Information Science in Shaanxi Normal University. Her research interests focus on applied mathematics and cryptography.

Yanjing Yang was born in 1995. She is currently pursuing the M.S. degree with School of Computer Science in Shaanxi Normal University. Her research interests focus on secure multi-party computation and information security.

Reviewers (Volume 21, 2019)

Dariush Abbasinezhad
Slim Abdelhedi
Mohd Faizal Abdollah
Ahmed Mohammed Abdullah
Subrata Acharya
Sodeif Ahadpou
Tohari Ahmad
Muhammad Najmi
Ahmad-Zabidi
Mohammad Reza Ahmadi
Asimi Ahmed
Mehrnaz Akbari Roumani
Abdul-Gabbar Tarish
Al-Tamimi
Aws N. Al-Zarqawee
Monjur M Alam
Shahid Alam
Tanweer Alam
Dilip S Aldar
Sara Ali
Ali Mohamed Allam
Khalid Abdulrazzaq
Alminshid
Ali Mohammed Alsahlany
Ruhul Amin
Rengarajan Amirtharajan
R. Anand
Karl Andersson
Benjamin Arazi
K. S. Arvind
Travis Atkison
Hany Fathy Atlam
Cossi Blaise Avoussoukpo
Anant M. Bagade
Amandeep Bagga
Nischay Bahl
Anuj Kumar Baitha
Saad Haj Bakry
R. R. Balakrishnan
Kavitha Balu
Maram Y Bani Younes
Tamer Mohamed Barakat
Utpal Barman
Pijush Barthakur
Eihab Bashier Mohammed
Bashier
Adil Bashir
Sunny Behal
Rydhm Beri
Taran Singh Bharati
Akashdeep Bhardwaj
Lathies T. Bhasker
Sugandh Bhatia
Sajal Bhatia
Krishna Bhowal
Sumitra Binu
Zhengjun Cao
Liling Cao
Chi-Shiang Chan
Eric Chan-Tin
Mohan Kumar Chandol
Yogesh Chandra
Arup Kumar Chattopadhyay
Nirbhay K. Chaubey
Ali M Chehab
Chi-Hua Chen
Tzung-Her Chen
Zhixiong Chen
Yi-Hui Chen
Chin-Ling Chen
Jan Min Chen
Qingfeng Cheng
Kaouthar Chetioui
Mao-Lun Chiang
Shu-Fen Chiou
Tae-Young Choe
Kim-Kwang Raymond Choo
Christopher P. Collins
Joshua C. Dagadu
Ashok Kumar Das
Prodipto Das
Sanjoy Das
Debasis Das
Ranjan Kumar Dash
Subhrajyoti Deb
Abdelrahman Desoky Desoky
Sankhanil Dey
Subhasish Dhal
Jintai Ding
Nishant Doshi
Ahmed Drissi
Qi Duan
Abd Allah Adel Elhabshy

Ahmed A. Elngar
Arizona Firdonsyah
Xingbing Fu
Vladimir Sergeevich Galyaev
Rakesh C Gangwar
Juntao Gao
Tiegang Gao
Xinwei Gao
G. Geetha
Mohammad GhasemiGol
Madhumala Ghosh
Ramesh Gopalan
Poornima Ediga Goud
Krishan Kumar Goyal
Ke Gu
Sumalatha Gunnala
Jatin Gupta
Charifa Hanin
Ali Hassan
Wien Hong
Tsung-Chih Hsiao
Defa Hu
Yen-Hung Hu
Xiong Hu
Chengyu Hu
Huajun Huang
Chin-Tser Huang
Jianmeng Huang
Munawar Hussain
Bala Venkateswarlu Isunuri
Grasha Jacob
Amit Jain
Yogendra Kumar Jain
Swati Jaiswal
Teena Jaiswal
V. S. Janani
N Jeyanthi
lin zhi jiang
Shaoquan Jiang
Rong Jiang
Rui Jiang
Zhengping Jin
Ashish Joshi
Omprakash Kaiwartya
Yoshito Kanamori
Nirmalya Kar
Gagandeep Kaur
Omar Khadir
Vaishali D. Khairnar
Asif Uddin Khan
Md. Al-Amin Khandaker
Malik Sikander Hayat Khiyal
Dong Seong Kim
P. Dhandapani Raman D. Kothandaraman
Anjan Krishnamurthy
Sajja Ratan Kumar
Manish Kumar
Naresh N Kumar
Saru Kumari
Yesem Kurt Peker
Then Lee
Yanping Li
Chun-Ta Li
Cheng Li
Zhaozheng Li
Chia-Chen Lin
Chih-Yang Lin
Iuon-Chang Lin
Yang-Bin Lin
Yining Liu
Shuang Gen Liu
Ximeng Liu
K. Shantha Kumari Luke Jayakumar
Zhiyong Luo
Ming Luo
Zahid Mahmood
Tanmoy Maitra
Arun Malik
T. Manesh
Ali Mansouri
Kamran Ali Memon
Weizhi Meng
Bo Meng
Yang Ming
Suhail Qadir Mir
Amit Mishra
anuranjan Misra
Madihah Mohd Saudi
Guillermo Morales-Luna
Belmekki Mostafa
Alaa Moualla
Hamdy M. Mousa
Kuntal Mukherjee
C. H. Mukundha
Bhagavathi Priya M Muthumanikam
Ambika Nagaraj
K. Nandhini
Syed Naqvi
Lakshmi Kannan Narnayanan

Sarmistha Neogy
Chokri Nouar
Abdul Abiodun Orunsolu
Nasrollah Pakniat
Dhiraj Pandey
B. D. Parameshachari
Subhash S. Parimalla
Chintan J. Patel
Kailas Ravsaheb Patil
Suresh Kumar Peddoju
Kanthakumar Pongaliur
A. Prakash
Munivara Prasad
Yudha Purwanto
Septafiansyah Dwi Putra
Murad Abdo Rassam Qasm
Qais Saif Qassim
Chuan Qin
Jiaohua Qin
Narasimhan Renga Raajan
Hashum Mohamed Rafiq
Abdul Hamid M. Ragab
V. Sampangi Raghav
Uma R. Rani
Golagani A.V.R.C Rao
Dhivya Ravi
Ramesh S Rawat
Siva Ranjani Reddi
Ou Ruan
Sanjay Kumar Sahay
Debabrata Samanta
Sabyasachi Samanta
Manju Sanghi
Arif Sari
Balamurugan K. S. Sathiah
Rajat Saxena
Michael Scott
Chandra Vorugunti Sekhar
Irwan Sembiring
Elena Sendroiu
Divyashikha Sethia
Vrutik M. Shah
Vrushank Shah
Kareemulla Shaik
Tarun Narayan Shankar
Udhayakumar Shanmugam
Rohith Shivashankar
Varun Shukla
Jitendra Singh
Debabrata Singh
Anuj Kumar Singh
Mahendra Pratap Singh
Bala Srinivasan
Siva Shankar Subramanian
Karthikeyan Subramanian
T. SudalaiMuthu
K. S. Suganya
Haiyan Sun
Maryam Tanha
Ariel Soares Teles
Pratik Teli
Xiuxia Tian
Geetam Singh Tomar
Yuan-Yu Tsai
Vandani Verma
Phu Vo Ngoc
Tao Wan
Fangwei Wang
Li Wang
Feng Wang
Libin Wang
Ying Wang
Ding Wang
Qingping Wang
Zhe Wei
C. H. Wei
Jianghong Wei
Na-I Wu
Degang Xu
Lei Xu
Chengbo Xu
Yashveer Yadav
Wei Yajuan
Li Yang
Wenjie Yang
Changsong Yang
Yifei Yao
Jun Ye
Pinghao Ye
Huang Yiwang
Lin You
Huifang Yu
Hang Yue
Dr Noor Zaman Zaman
Sherali Zeadally
Jianping Zeng
Jie Xiu Zhang
Qiu-Yu Zhang
Yanshuo Zhang
Fangguo Zhang
Zonghua Zhang
Futai Zhang

Yinghui Zhang
Jianhong Zhang
Hongzhuan Zhao
Zhiping Zhou
Ye Zhu
Yingwu Zhu
Frank Zhu
Aaron Zimba