IJNS
International Journal of Network Security

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

Vol. 24, No. 6 (November 2022)

INTERNATIONAL JOURNAL OF NETWORK SECURITY

PUBLISHING OFFICE

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

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

PUBLISHER: Candy C. H. Lin

Volume: 24, No: 6 (November 1, 2022)

International Journal of Network Security

A Hybrid Iterative Greedy Optimization Algorithm for Distributed Assembly Blocking Flow Shop Scheduling Problem

Yongqi Zheng

(Corresponding author: Yongqi Zheng)

Zhengzhou Shuqing Medical College
Zhengzhou 450000, China
Email: zhengyongqi666@sohu.com

(Received Nov. 20, 2021; Revised and Accepted Oct. 11, 2022; First Online Oct. 22, 2022)

Abstract

The distributed shop scheduling problem is to study the distribution of jobs between factories and the processing sequence of each factory in the context of collaborative production to optimize a particular index. This paper proposes a hybrid iterative greedy optimization algorithm (HIGOA) to address the distributed assembly blocking flow shop scheduling problem (DABFSP). The improved Nawaz-Enscore-Ham (NEH) rule is utilized to sort all the jobs according to the total processing time in the initialization stage. A feasible scheduling sequence is constructed as the initial solution of the algorithm. The discrete migration and butterfly adjustment operators are used for iterative updating to find a superior candidate solution for the sub-generation. The destructively refactoring mechanism is introduced to randomly remove elements from existing solutions and construct new sequences by re-inserting the deleted elements through a greedy selection process. The embedded local search strategy aims to insert the feasible iterative sequence into the optimal neighborhood location, thereby improving the algorithm's accuracy. HIGOA and eight state-of-arts algorithms for solving distributed assembly scheduling problems are tested on 900 problem instances. Experimental results show the efficient effectiveness of HIGOA in solving DABFSP.

Keywords: Discrete Butterfly Operator; Distributed Assembly Blocking Flow Shop Scheduling; Embedded Local Search Strategy; Hybrid Iterative Greedy Algorithm

1 Introduction

In the context of economic globalization, distributed manufacturing has gradually become one of the common production modes in the international manufacturing industry. Distributed assembly blocking flow shop scheduling problem is a typical NP-hard combinatorial optimization problem [13]. It is an important field in multiple distributed production systems [25]. Compared with traditional single-factory manufacturing, distributed production and manufacturing can integrate the resources of multiple manufacturing enterprises or factories, arrange production tasks or specific production indicators, improve the comprehensive manufacturing efficiency of enterprises and quickly respond to market changes [8]. The distributed shop scheduling problem studies the processing sequence of multiple factory jobs to satisfy a certain index of optimization [29]. Therefore, a variety of distributed production scheduling problems have attracted more and more researchers' attention, such as distributed permutation flow shop scheduling problem (DPFSP) [1], distributed blocking flow shop scheduling problem (DBFSP) [4, 18], distributed parallel machine processing shop scheduling problem (DPMPSP) [21], etc.

With distributed manufacturing as the background, multi-factory cooperative production mode is increasingly popular, and the cooperation between factories or enterprises and other manufacturing enterprises has become one of the strategic goals of sustainable development. The theoretical analysis and optimization of distributed flow shop scheduling problem in multiple production centers is the research hotspot in this field [10]. The intelligent optimization algorithms efficiently solve the actual production demand problem compared with the traditional method. At the same time, intelligent optimization algorithms are widely concerned when they are used to solve network security related problems [28].

With the continuous development and popularization of Internet technology, the types of applications and services in the network continue to increase, and traditional methods can no longer meet the processing of large data volumes. Numerous intelligent optimization algorithms are applied to solve network security related [22, 24]. The intelligent optimization algorithm can realize the efficient sharing of information resources by comprehensively and

effectively analyzing the security of the computer network system[13].

An iterative greedy (ORIG) and a discrete Artificial bee colony (ORABC) algorithm with two-stage search and multi-neighborhood search mechanism is proposed by Meng *et al.*[14] to solve the DPFSP with order constraints. A neighborhood search based on sequential insertion and a greedy re-insertion strategy are designed, and a two-level search strategy is introduced to improve the search performance of the algorithm. A pareto-based distribution estimation algorithm (PEDA) is proposed by Shao *et al.* [16] to solve the DNWFSP. The PWQ heuristic is extended to the distributed environment to generate initial individuals, and a sampling method with parameter templates is proposed to generate offspring individuals. Several proposed neighborhood search methods are utilized to optimize the candidate solutions. Three discrete invasive weed optimization (DIWO) algorithms are proposed by Sang *et al.* [15] to solve the distributed assembly replacement flow shop scheduling problem. Neighborhood design based on product arrangement and job sequence is designed by combining the knowledge of specific problems and the idea of weed invasion to improve the global search ability of the algorithm.

DABFSP is divided into two stages: processing stage and assembly stage. A series of jobs are processed on machines in multiple specific factories (the processing stage) and then assembled by one machine into the final product (the assembly stage). The jobs in each processing factory are assigned to a blocking flow shop with isomorphic parallel machines in the processing stage, and no intermediate buffer exists between adjacent machines in the factory. One of the key problems to be solved is to prioritize the jobs to minimize the processing time. The finished jobs are transferred to the assembly shop in the assembly stage, and a machine is used to assemble the finished jobs into the final product. The distributed permutation flow shop scheduling and assembly scheduling are combined in the problem. The goal is to find the optimal scheduling sequence under the minimum maximum assembly completion time.

A collaborative water wave optimization algorithm for distributed assembly no-idle flow shop scheduling problem (DANIFSP) is proposed to minimize the maximum assembly completion time [26]. The reinforcement learning mechanism based on VNS framework is designed, and an improved fracture operator combined with path relinking and VNS method is introduced. These two operations enhance the exploration and development ability of the algorithm. In the refraction phase, the multi-neighborhood perturbation strategy is used to extract knowledge information to escape from the local optimal solution. Experimental results show that this algorithm is stable and effective. A distributed two-stage assembly flow scheduling problem is proposed to obtain the minimum completion time [23]. Three hybrid meta-heuristic algorithms are proposed and the parameters of HVNS are adjusted by one-way an OVA. Experimental results show that HGA-RVNS has significant advantages in large-scale problems. A distributed assembly permutation flow shop scheduling problem is proposed to minimize the maximum completion time of the entire manufacturing process [7]. The problem consists of two stages: the processing stage in the production factory and the assembling stage of the jobs into the product in the assembly factory. A meta-heuristic algorithm using bias random iterative local search without parameters is proposed. A large number of experimental results show that this algorithm has better performance than other algorithms. An optimal block knowledge-driven backtracking search algorithm (BKBSA) is proposed to solve distributed assembly flow-shop scheduling problem with minimum assembly process completion time [27]. Three constructive heuristic methods are proposed to generate competitive initial solutions. A mutation strategy based on block movement is used to ensure that the optimal subsequence of the candidate solution is not destroyed in the mutation operation. The similarity between candidate solutions is used as feedback index to control the utilization of block shift. The BKBSA algorithm is compared with the other three algorithms on 810 test instances, and the experimental results show that BKBSA is a competitive algorithm. In addition, various studies on other distributed assembly scheduling problems have achieved certain results [5, 9].

Iterated Greedy (IG) [14] algorithm is a simple and efficient heuristic algorithm with fast calculation speed, few control parameters and excellent optimization effect. IG algorithm searches the solution space of the problem through destruction and reconstruction operations, which makes IG have strong global search ability. However, the condition of insufficient mining depth exists in the face of different complex optimization problems [3]. Monarch butterfly optimization (MBO) [20] algorithm is a novel swarm intelligence optimization algorithm with excellent performance in many optimization problems. The monarch optimization algorithm is prone to problems such as reduced population diversity and weak global search ability in the migration process [2]. Therefore, to integrate the IG algorithm with the traditional MBO algorithm is of great research value, because IG focuses on searching the solution space from the global level and has fast calculation speed. A hybrid iterative greedy algorithm (HIGOA) is proposed according to the problem characteristics of DABFSP to optimize the maximum assembly completion time. The monarch butterfly optimization (MBO) migration operator and butterfly adjustment operator are introduced to enhance the global search ability of MBO, so that satisfactory candidate solutions are found in a reasonable time. The contributions are as follows.

The mixed integer linear programming model (MILP) of DABFSP is designed. HIGOA is designed to solve the DABFSP. The destruction, construction and two discretized butterfly operators are introduced to update the population in the iterative stage. The embedded local search strategy is designed to improve the accuracy of the

algorithm. The rest of the article is organized as follows. The distributed assembly blocking flow shop scheduling problem is presented in Section 2. The hybrid iterative greedy algorithm is introduced in Section 3. The experiment and results are designed in Section 4. The Section 5 gives the conclusion.

2 The Distributed Assembly Blocking Flow Shop Scheduling Problem

2.1 Symbol Representation and Problem Description

The distributed assembly blocking flow shop scheduling problem is described as follows: in the production phase, n jobs $\{J_1, J_2, \cdots, J_n\}$ are processed in F factories $\{F_1, F_2, \cdots, F_F\}$, where each factory has the same m machines $\{M_1, M_2, \cdots, M_m\}$, and each factory is a blocking flow shop. In the assembly stage, n jobs to be assembled $\{J_1, J_2, \cdots, J_n\}$ is assembled into s products $\{P_1, P_2, \cdots, P_S\}$. Only one assembly factory F_A and only one assembly machine M_A exist in the assembly factory as shown in Figure 1. Product $P = \{P_1, P_2, \cdots, P_S\}$ consists of N_h jobs to be machined, and all of them satisfy $\sum_{h=1}^{s} N_h = n$. The jobs to be machined are transferred to N_h jobs to be assembled belonging to product h if and only when all N_h jobs to be machined have completed the process.

Each job is assigned to any factory, but once the job is assigned to a factory, it can only be processed at that factory. The processing time of the job on a particular machine is predetermined regardless of the factory to which it is assigned. No intermediate buffer exists between any two consecutive machines in each factory. In other words, the job in process is not allowed to leave the current machine until the next machine starts processing. In addition, each machine can process a maximum of one job at a time, and each job can be processed by a maximum of one machine. Once the machine starts processing, each job must be completed without interruption, and preemption is not allowed. If all jobs belonging to product P_h are completed in the processing phase and M_A is idle in the assembly factory, the assembly process begins.

2.2 The Mixed Integer Linear Programming Model of DABFSP

In this paper, the optimization objective is to solve the minimum maximum assembly completion time $\min C_{max}(\pi)$. The constraints of DABFSP to meet are as follows.

$$\sum_{f=1}^{F}\sum_{k=1}^{n} x_{i,k,f} = 1, i = 1, 2, \cdots, n \quad (1)$$

$$\sum_{i=1}^{n} x_{i,k,f} \leq 1, \quad k = 1, 2, \cdots, n \ \&f = 1, 2, \cdots, F \quad (2)$$

where $x_{i,k,f} \in \{0, 1\}$ represents the decision variable, and Equation (1) ensures that each job to be processed can only be assigned to one factory and only appear at one position in the feasible scheduling sequence of the factory. Equation (2) ensures that only one job to be machined is assigned to each factory location.

$$D_{f,k,0} \geq D_{f,k-1,1}, \quad k = 1, 2, \cdots, n \ \&f = 1, 2, \cdots, F \quad (3)$$

$$D_{f,k,j} \geq D_{f,k,j-1} + \sum_{i=1}^{n} x_{i,k,f} \cdot p_{i,j}, k = 1, 2, \cdots, n, \quad j = 1, 2, \cdots, m, \ \&f = 1, 2, \cdots, F \quad (4)$$

$$D_{f,k,j} \geq D_{f,k-1,j+1}, k = 2, 3, \cdots, n, \quad j = 1, 2, \cdots, m-1, \ \&f = 1, 2, \cdots, F \quad (5)$$

$$D_i \geq D_{f,k,m} - L \star (1 - x_{i,k,f}), k = 1, 2, \cdots, n, \quad i = 1, 2, \cdots, n, \ \&f = 1, 2, \cdots, F \quad (6)$$

Equation (3) represents the start time of the job to be processed on the first machine in each factory. Equation (4) represents the relationship between the departure time of two adjacent jobs in each factory. Equation (5) represents the relationship between the departure time of two adjacent processed job in each factory, indicating that the blocking constraint is satisfied. Equation (6) determines the time at which each job leaves the processing plant.

$$DA_h \geq DA_{h-1} + t_h, h = 1, 2, \cdots, S \quad (7)$$

$$DA_h \geq D_i + t_h - L \star (1 - G_{i,h}), \quad i = 1, 2, \cdots, n, \& h = 1, 2, \cdots, S \quad (8)$$

$$C_{max} \geq DA_h, h = 1, 2, \cdots, S \quad (9)$$

$$DA_h > 0. \quad (10)$$

Equations (7) & (8) ensure the relationship between adjacent assembly products and that each product is assembled only after all generation jobs have been processed. Equations (9) & (10) define the maximum assembly completion time, and the assembly completion time of each product must be greater than 0.

3 Hybrid Iterative Greedy Algorithm

Iterative greedy algorithm (IG) is a simple and efficient constructive heuristic algorithm. IG is divided into two phases: destruction and reconstruction. Some solutions are removed from previously built complete candidate solutions and re-constructed complete candidate solutions in a greedy and constructive manner. Monarchs butterfly optimization algorithm (MBO), which is a meta-heuristic algorithm based on swarm intelligence, is directly used

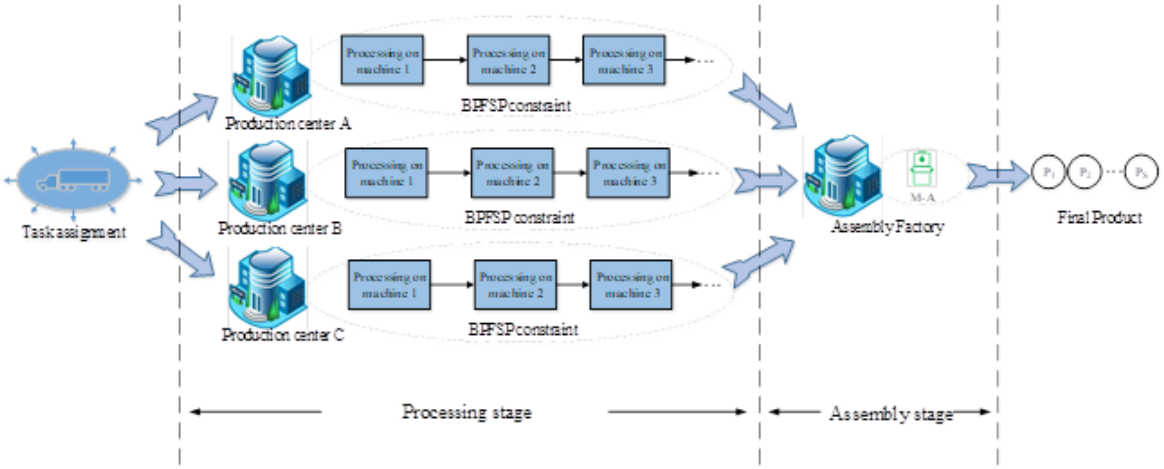


Figure 1: Schematic diagram of distributed assembly blocking flow shop scheduling

to solve continuous optimization problems. Various optimization problems are solved by studying and simulating monarch butterfly migration behavior in nature in MBO. Therefore, a hybrid iterative greedy algorithm, combined with the characteristics of distributed blocking flow shop scheduling problem, is proposed in this paper. The migration operator and butterfly adjustment operator of MBO algorithm are discretized and embedded into HIGOA.

Table 1: Initialize job setup

Product	Job	Process time			Assembly time
		m_1	m_2	m_3	m_A
P_1	2	4	2	2	3
	3	1	3	2	
	5	2	2	2	
P_2	1	1	3	6	2
	4	4	2	3	

3.1 Initialization, Destruction, and Reconstruction

The improved NEH rules are utilized to arrange all the jobs to be processed in the HIGOA initialization stage. The jobs of the same product are preferentially allocated to machines in different factories for processing considering the total processing time T_i. Product P_1 contains the job sequence [4] and product P_2 contains the artifact sequence 1,4 as shown in Table 1. Assuming that the factory processes only all the job belonging to product P_1 or product P_2 at this time, the first job of each factory is determined and the product is inserted at the position with the minimum total processing completion time among all possible neighborhood locations until all products are considered. Taking the job processing time initialized in Table 1 as an example, a set of initialization sequences are obtained by the above steps, which is used as the initialization population of HIGOA algorithm in Figure 2.

The algorithm starts the destruction and reconstruction phase to provide diversified solutions after the initial population construction is completed.

Destruction stage: d unduplicated jobs are randomly selected to be deleted from the sequence and a new sequence π_1 is formed, that is, the sequence after d jobs are deleted. In addition, the sequence of d deleted jobs are represented as π_d.

Reconstruction stage: the jobs of sequence π_d is taken out and inserted into the sequence π_d. Firstly, the first job of π_d is taken out and inserted into all possible positions of π_d until the minimum $\min C_{max}(\pi)$ is generated in π_d. This process keeps iterating until π_d is empty.

3.2 Discrete Migration Operator and Butterfly Adjustment Operator

Migration operator and adjustment operator are important operators in MBO, which help to improve the global search ability of the algorithm and avoid falling into local optimum. In this chapter, the migration operator and adjustment operator are discretized and embedded into HIGOA algorithm.

The current population is divided into two parts: subpopulation a and subpopulation b according to the sorting results after destruction reconstruction. Discrete migration operator (DMO) is used to update offspring in population a. Population b uses the discrete butterfly adjustment operator (DBAO) to generate the remaining candidate solutions. So NP_a is the number of monarchs in population a, and NP_b is the number of monarchs in population b.

$$NP_a = \lceil p \star N \rceil \quad (11)$$

$$NP_b = N - NP_a \quad (12)$$

where N indicates population size and $P = \frac{7}{12}$ indicates the mobility of monarchs in population a. New candidate

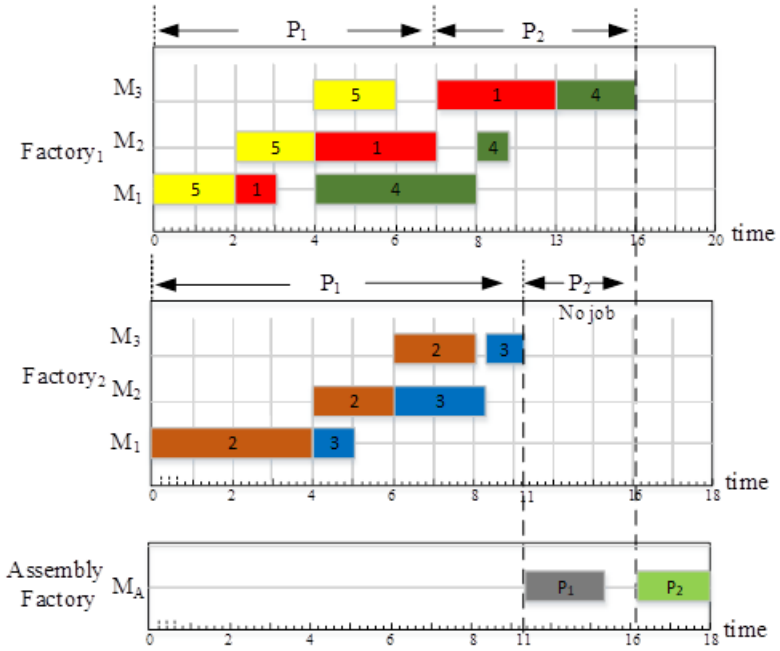


Figure 2: A Gantt chart for the example instance

solutions are generated as follows in subpopulation:

$$x_t^{g+1} \leftrightarrow \begin{cases} x_{r_1}^g & \text{if } r \le p \\ x_{r_2}^g & \text{elsewise} \end{cases} \quad (13)$$

$$r = rand \star peri \quad (14)$$

where g is the current generation HIGOA, x_t^{g+1} is the t-th candidate solution in iteration $g+1$, $x_{r_1}^g$ is the r_1 candidate solution in g generation, $x_{r_2}^g$ is the r_2 candidate solution in g generation. r represents a random number, as shown in Equation (14). peri is set to 1.2. r_1 and r_2 represent subscripts of randomly selected individuals in subpopulations a and b, respectively. The sign $\leftrightarrow$ means exchange. New candidate solutions in subpopulation b are generated as follows:

$$x_k^{g+1} \leftrightarrow \begin{cases} x_{best}^g & \text{if } rand \le p \\ x_{r_3}^g & \text{elsewise} \end{cases} \quad (15)$$

where, x_k^{g+1} is the k-th candidate solution when iterating $g+1$, x_{best}^g is the optimal candidate solution of the g generation of subpopulations a and b. $x_{r_3}^g$ is the r_3 candidate solution when iterating g. rand means a random number ranging from [0,1], and the symbol $\leftrightarrow$ signifies exchange.

3.3 Local Search Strategy

The precision search ability is necessary to be enhanced to find the optimal solution quickly in the later iteration of the algorithm. The local search strategy is helpful to improve the exploitation ability of the algorithm, so it is a crucial step to increase the local search ability of the algorithm. The sequence of candidate solutions generated by the above steps is reinserted at all possible locations in the factory until the local search is terminated if no better solution is found. Finally, the position that minimizes the maximum assembly completion time after inserting the job is selected as the final position of the job, as shown in Figure 3.

3.4 Algorithm Flow

HIGOA focuses on improving the global search ability in the early exploration stage of the algorithm, while HIGOA focuses on improving the local search ability of the algorithm in the later iteration. Exploration and exploitation capabilities are effectively balanced by algorithms using destruction and reconstruction, discrete migration operators and butterfly adjustment operators, and local search strategies. The pseudocode of HIGOA algorithm proposed in this paper is shown in Algorithm 1.

4 Experiment Design and Result Analysis

4.1 Experiment Design

The performance of HIGOA is tested on 900 instances in this study. The test set is composed of different quantities of factories, machines, jobs and products. The combination of examples is shown in Table 2. The same experimental conditions were used in the experiment, and all algorithms were programmed by MATLAB to ensure the fairness of the algorithm. Meanwhile, the performance of HIGOA is compared with that of the newer H2NRa [17],

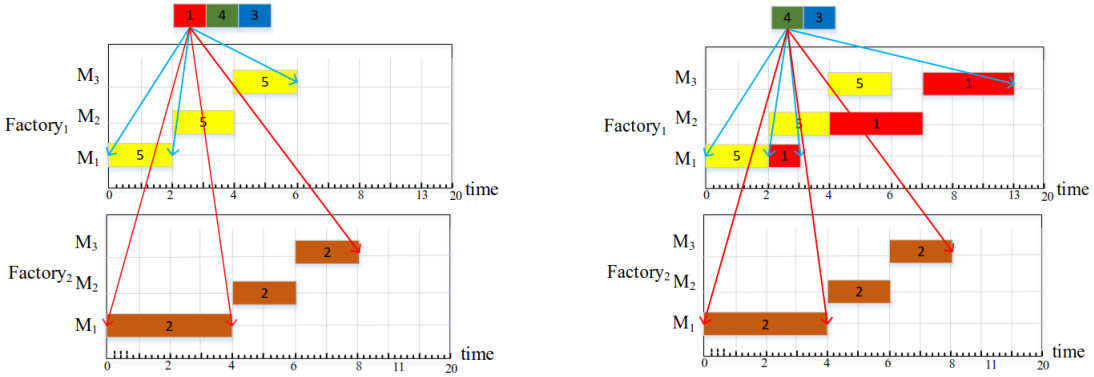


Figure 3: Local search strategy image

Algorithm 1 The pseudocode of HIGOA

Algorithm 1 The pseudocode of HIGOA

Input: Test function
Initialize g, D, N, $MaxGen$
Use NEH initialization method to construct the population
$g = 1$
while ($g < MaxGen$)
Destruct the sequence of the population
Reconstruct the sequence of the population
  Divide the population into two subpopulations
    Update the subpopulation a **using discrete migration operator**
    Update the subpopulation b **using discrete adjustment operator**
  Merge two subpopulations **using the local search mechanism**
Calculate the fitness value, $t = t + 1$
end while
Output C_{max}

HFL, HNEH, HMME, HLPT [19], H11,H12,H21 [12] competitive algorithms in the literature.

Average relative percentage deviation (ARPD) is used to measure the experimental results in this experiment, as shown below:

$$ARPD = \frac{1}{Run} \star \sum_{r=1}^{Run} \frac{C_t^r - C_{g_{best}}}{C_{g_{best}}} \star 100\% \qquad (16)$$

where, C_t^r is the result of the t-th algorithm of the current instance, and run is the number of times the algorithm runs independently. $C_{g_{best}}$ is the best candidate solution found so far. Monte Carlo method is used in the experiment, and all the examples are independently run for 21 times. The experimental results are shown in Table 3 and Figure 4. As shown in Table 3, the HIGOA algorithm is superior to any other comparison algorithm. So we randomly selected several algorithms (H2NRa, HFL, HLPT, H11) for comparison.

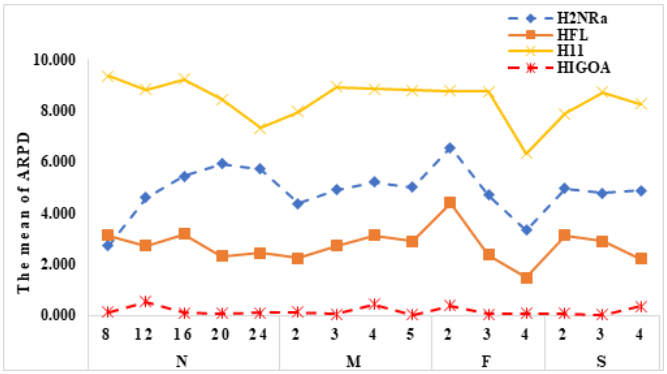


Figure 4: Visualization of experimental results

4.2 Result Analysis

To get a more intuitive view of HIGOA's performance compared to other comparison algorithms, Figure 5 shows the probability graph for 95% confidence intervals compared to other comparison algorithms. In a probability graph, each set of data contains three lines, with the middle line representing the fitted distribution of the data and the dashed lines on both sides representing confidence intervals. If a set of data is closely distributed along the median line and near the left side of the graph, the algorithm is superior to other algorithms. It can be seen from the probability graph that the experimental results of HIGOA algorithm are optimal among all experimental algorithms when normal distribution is satisfied.

Wilcoxon test and Friedman test are used to analyze the statistical differences of experimental results to further verify the differences between HIGOA algorithm and other comparison algorithms. In Wilcoxon rank-sum test, if the p-value is less than α, it indicates that there is a significant difference between the two algorithms, as shown in Table 4. Pairwise comparison of HIGOA with the other four algorithms shows that all p-values are less than α. Therefore, HIGOA performs significantly better than other comparison algorithms in solving DAPFSP problems.

Table 2: Combination of examples

			n				m				f			s	
Number	8	12	16	20	24	2	3	4	5	2	3	4	2	3	4

Table 3: ARPD results

			H2NRa	HFL	HMME	HNEH	HLPT	H11	H12	H21	HIGOA
1		8	2.725	3.133	2.702	2.857	2.646	9.368	9.659	4.223	0.131
2		12	4.606	2.738	2.311	2.245	2.201	8.819	8.985	3.511	0.543
3	n	16	5.448	3.186	2.72	2.952	2.714	9.230	8.265	4.3	0.120
4		20	5.944	2.320	2.221	2.136	1.939	8.460	7.815	3.932	0.087
5		24	5.732	2.442	2.145	2.212	1.963	7.322	6.723	3.632	0.100
6		2	4.370	2.236	1.724	2.157	1.461	7.957	7.878	3.711	0.123
7	m	3	4.915	2.733	2.521	2.458	2.560	8.932	8.261	4	0.066
8		4	5.225	3.141	2.812	2.721	2.449	8.849	7.787	4.235	0.435
9		5	5.006	2.919	2.321	2.212	2.681	8.809	7.226	3.454	0.030
10		2	6.562	4.409	3.621	3.923	3.983	8.782	9.522	5.645	0.397
11	f	3	4.723	2.382	2.355	2.2	1.870	8.766	7.722	3.721	0.071
12		4	3.348	1.474	1.454	1.1	1.002	6.344	5.369	2.521	0.072
13		2	4.972	3.133	2.554	2.754	2.345	7.888	7.811	5.211	0.073
14	s	3	4.777	2.916	2.321	2.345	2.543	8.737	7.023	3.522	0.045
15		4	4.884	2.215	2.235	2.357	1.967	8.267	7.811	3.002	0.372
	Mean		4.882	2.758	2.401	2.442	2.288	8.435	7.857	3.908	0.178

Table 4: Wilcoxon result

HIGOA vs	R^+	R^-	Bonding Value	p-value	$\alpha = 0.05$	$\alpha = 0.1$
H2NRa	772	12	116	1.4784E-127	yes	yes
HFL	533	26	341	1.4078E-87	yes	yes
HLPT	431	26	419	2.1736E-68	yes	yes
H11	833	3	64	3.7455E-136	yes	yes

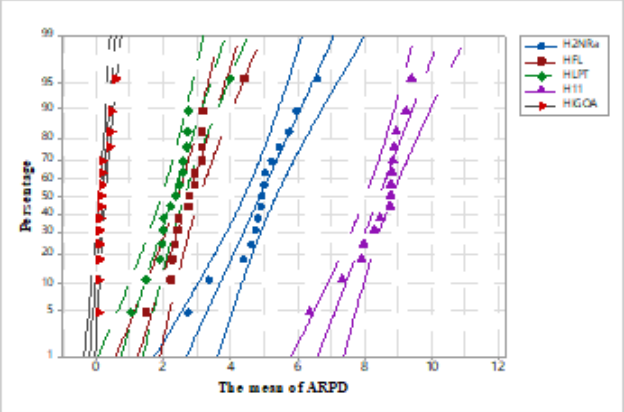


Figure 5: Probability graph of results

Friedman tests are statistical tests used to determine differences between multiple (related) samples that reflect performance differences between all algorithms. Friedman test is performed on HIGOA and other four comparison algorithms according to the number of factories and jobs to be processed, and the results are shown in Figure 6. Under 95% confidence interval, HIGOA algorithm has better competitive performance than other comparison algorithms.

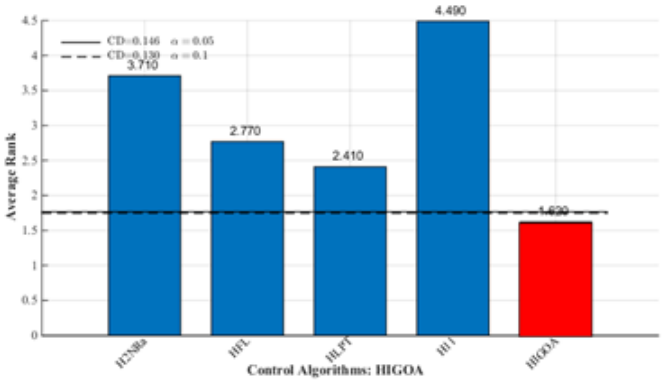


Figure 6: The results of the Friedman-test

5 Conclusion

A hybrid iterative greedy algorithm (HIGOA) is proposed to solve the distributed assembly blocking flow shop scheduling problem. The HIGOA algorithm combines the characteristics of DABFSP and constructs promising initial candidate solutions using the improved NEH in the initialization stage of the algorithm. In the iterative stage, destruction-reconstruction and two cooperative butterfly discrete operators are used to ensure that the candidate solution escapes local optimum in the iterative search stage. Finally, the embedded local search strategy is used to improve the accuracy of candidate solutions. The visualization experiment results and statistical analysis show that HIGOA algorithm has the advantages of higher precision and higher stability than contrast algorithm. The hybrid iterative greedy algorithm HIGOA proposed in this paper provides a new solution for DAPFSP. HIGOA algorithm has a significant advantage in solving 900 problem instances, and there is a large research space for its application to distributed flow shop scheduling problems with other constraints. In the future, distributed flow shop scheduling problems with other constraints will be considered. In addition, it would be very interesting if HIGOA is applied to network security related research such as color image encryption and network intrusion analysis.

Acknowledgments

The author would like to thank the anonymous reviewers for their valuable comments and suggestions, which have improved the presentation of this paper. This research is supported by the Social Sciences in the Federation of Henan Provincial, China (SKL-2019-264), the subject of Zhengzhou Shuqing Medical College, China (JX20180149).

References

[1] A. Ali, Y. Gajpal,, and T. Y. Elmekkawy, "Distributed permutation flowshop scheduling problem with total completion time objective," *OPSEARCH*, vol. 58, no. 2, pp. 425-447, 2021.

[2] M. Alweshah, "Solving feature selection problems by combining mutation and crossover operations with the monarch butterfly optimization algorithm," *Applied Intelligence*, vol. 51, pp. 4058-4081, 2021.

[3] S. Chen, Q. K. Pan and L. Gao, "Production scheduling for blocking flowshop in distributed environment using effective heuristics and iterated greedy algorithm," *Robotics and Computer-Integrated Manufacturing*, vol. 71, p. 102155, 2021..

[4] S. Chen, Q. K. Pan, L. Gao, and H. Y. Sang, "A population-based iterated greedy algorithm to minimize total flowtime for the distributed blocking flowshop scheduling problem," *Engineering Applications of Artificial Intelligence*, vol. 104, pp. 104375, 2021.

[5] Y. Y. Huang, Q. K. Pan, J. P. Huang, P. N. Suganthan and L. Gao, "An improved iterated greedy algorithm for the distributed assembly permutation flowshop scheduling problem," *Computers and Industrial Engineering*, vol. 152, p. 107021, 2020.

[6] Y. Jian, Peng, L. Jian and X. Y. Dong, " Research on network intrusion detection based on improved machine learning method," *International Journal of Network Security*, vol. 24, no. 3, PP.533-540, 2022.

[7] D. Ferone, S. Hatami, E. M. González-Neira, et al., "A biased-randomized iterated local search for the

distributed assembly permutation flow-shop problem," *International Transactions in Operational Research*, vol. 27, 2020.

[8] J. P. Huang, Q. K. Pan, Z. H. Miao, and L. Gao, "Effective constructive heuristics and discrete bee colony optimization for distributed flowshop with setup times," *Engineering Applications of Artificial Intelligence*, vol. 97, pp. 104016, 2021.

[9] J. Y. Mao, Q. K. Pan, Z. H. Miao and L. Gao, "An effective multi-start iterated greedy algorithm to minimize makespan for the distributed permutation flowshop scheduling problem with preventive maintenance," *Expert Systems with Applications*, vol. 169, p. 114495, 2020.

[10] T. Meng and Q. K. Pan, "A distributed heterogeneous permutation flowshop scheduling problem with lot-streaming and carryover sequence-dependent setup time," *Swarm and Evolutionary Computation*, vol. 60, p. 100804, 2021.

[11] T. Meng, Q. K. Pan and L. Wang, "A distributed permutation flowshop scheduling problem with the customer order constraint," *Knowledge-Based Systems*, vol. 184, pp. 104894.1-104894.17, 2019.

[12] Q. K. Pan, L. Gao, X. Y. Li, et al., "Effective constructive heuristics and meta-heuristics for the distributed assembly permutation flowshop scheduling problem," *Applied Soft Computing*, vol. 81, p. 105492, 2019.

[13] Q. K. Pan, L. Gao, and L. Wang, "An effective cooperative co-evolutionary algorithm for distributed flowshop group scheduling problems," *IEEE Transactions on Cybernetics*, vol. 52, no. 7, pp. 5999-6012, July 2022.

[14] R. Ruiz and T. Stützle, "A simple and effective iterated greedy algorithm for the permutation flowshop scheduling problem," *European Journal of Operational Research*, vol. 177, pp. 2033-2049, 2007.

[15] H. Y. Sang, Q. K. Pan, J. Q. Li, P. Wang, Y. Y. Han, K. Z. Gao, P. Duan, "Effective invasive weed optimization algorithms for distributed assembly permutation flowshop problem with total flowtime criterion," *Swarm and Evolutionary Computation*, vol. 44, pp. 64-73, 2019.

[16] W. S. Shao, D. C. Pi, and Z. S. Shao, "A pareto-based estimation of distribution algorithm for solving multiobjective distributed no-wait flow-shop scheduling problem with sequence-dependent setup time," *IEEE Transactions on Automation Science and Engineering*, vol. 16, no. 3, pp. 1344-1360, July 2019.

[17] W. S. Shao, D. C. Pi, and Z. S. Shao, "Local search methods for a distributed assembly no-idle flow shop scheduling problem," *IEEE Systems Journal*, vol. 13, no. 2, pp. 1945-1956, June 2019.

[18] Z. Shao, W. Shao, D. Pi, "Effective heuristics and metaheuristics for the distributed fuzzy blocking flow-shop scheduling problem," *Swarm and Evolutionary Computation*, vol. 59, 2020.

[19] Z. S. Shao, W. S. Shao, D. C. Pi, "Effective constructive heuristic and metaheuristic for the distributed assembly blocking flow-shop scheduling problem," *Applied Intelligence*, vol. 50, 2020.

[20] G. G. Wang, S. Deb, X. Zhao, and Z. Cui, "A new monarch butterfly optimization with an improved crossover operator," *Operational Research*, vol. 18, pp. 731–755, 2016.

[21] M. Wang and G. H. Pan, "A novel imperialist competitive algorithm with multi-elite individuals guidance for multi-object unrelated parallel machine scheduling problem," *IEEE Access*, vol. 7, pp. 121223-121235, 2019.

[22] F. Wei, H. J. Gao and D. Cao "Performance study of a network intrusion detection algorithm improved by an optimization algorithm," *International Journal of Network Security*, vol. 24, no. 5, PP.953-958, 2022.

[23] F. Xiong, K. Xing, W. Feng, L. Hang and L. Han, "Minimizing the total completion time in a distributed two stage assembly system with setup times," *Computers and Operations Research*, vol. 47, pp. 92-105, 2014.

[24] L. Yan, X. W. Wang, and S. L. Yin, " Campus garbage image classification algorithm based on new attention mechanism," *International Journal of Electronics and Information Engineering*, vol. 13, no. 4, 2021, pp. 131-141, 2021.

[25] F. Q. Zhao, X. He, and L. Wang, "A two-stage cooperative evolutionary algorithm with problem-specific knowledge for energy-efficient scheduling of no-wait flow-shop problem," *IEEE Transactions on Cybernetics*, vol. 51, no. 11, pp. 5291-5303, Nov. 2021.

[26] F. Q. Zhao, L. Zhang, J. Cao, and J. Tang, "A cooperative water wave optimization algorithm with reinforcement learning for the distributed assembly no-idle flowshop scheduling problem," *Computers and Industrial Engineering*, vol. 153, p. 107082, 2020.

[27] F. Q. Zhao, J. L. Zhao, L. Wang, *et al.*, "An optimal block knowledge driven backtracking search algorithm for distributed assembly No-wait flow shop scheduling problem," *Applied Soft Computing*, vol. 112, pp. 107750, 2021.

[28] N. Zhao, X. W. Wang, and S. L. Yin, "Research of fire smoke detection algorithm based on video," *International Journal of Electronics and Information Engineering*, vol. 13, pp. 1-9, 2021.

[29] J. Zheng, L. Wang, and J. J. Wang, "A cooperative coevolution algorithm for multi-objective fuzzy distributed hybrid flow shop," *Knowledge-Based Systems*, vol. 194, pp. 105536, 2020.

Biography

Zheng Yongqi Biography. Zheng Yongqi, female, was born in Henan Province, China in Novemer 1984. She is a lecturer in Zhengzhou Shuqing Medical College. Research direction: Information security.

A Dynamic Trust Evaluation Model of User Behavior Based on Transformer

Xiuwen Yu[1], Rong Huang[1], Yuancheng Li[1], Rixuan Qiu[2], Xin Zhou[2], Liang Liang[2], and Sitong Jing[3]
(Corresponding author: Yuancheng Li)

School of Control and Computer Engineering & North China Electric Power University[1]
No. 2 Beinong Road, Changping District, Beijing, China
Information & Telecommunication Branch of State Grid Jiangxi Electric Power Supply Co., Ltd.[2]
No. 7077 Changdong Avenue, High-tech Zone, Nanchang, Jiangxi Province
PowerChina Jiangxi Electric Power Engineering Co., Ltd.[3]
No. 426, Jingdong Avenue, Qingshanhu District, Nanchang City, Jiangxi Province
Email: ncepua@163.com
(Received Apr. 20, 2022; Revised and Accepted Oct. 12, 2022; First Online Oct. 15, 2022)

Abstract

In the boundary-based protection system, although user authentication can provide a certain degree of security, when user information is leaked, this method will be difficult to deal with the attacking threat from internal and external legitimate users. Aiming at the attacking threat of legitimate users, we analyze the user behavior and constructs a dynamic trust evaluation model of user behavior based on the Transformer network. Firstly, unsupervised pre-training of the Transformer network and extracting the time characteristics of data can effectively improve the model's generalization ability. Then, fine-tune the network parameters, build the trust model of users' historical behavior, predict a user's future behavior through the established trust model, and calculate the similarity between the predicted behavior and users' actual behavior to evaluate the trust of users' behavior. Finally, we design and conduct experiments on a public data set. The experimental results prove the effectiveness of this method and also show that this method can reduce the training time and improve the accuracy of trust evaluation.

Keywords: Transformer Network; Trust Evaluation; Unsupervised Pre-training; User Behavior

1 Introduction

In the Internet of things environment, with the extensive access of a large number of terminal equipment and users, the network exposure is increasing, which brings severe challenges to the existing protection system [4]. However, the authentication and access control of the existing IoT terminal equipment and users mostly adopt the method of "one-time authentication, one-time authorization and long-term effectiveness." After the authentication is passed, it has legal authority for a long time and can carry out any operation within the scope of authority. Due to the lack of continuous behavior analysis, authentication and access control measures, it is impossible to solve the problem that legitimate terminal devices or users are illegally controlled by attackers to access the company's data and business resources in a legitimate capacity. At the same time, for insiders, due to the preset trust mechanism of insiders, if insiders conduct illegal operations or launch malicious attacks, it is difficult to control effectively and will cause huge losses.

With the gradual expansion of the scale of users in the Internet of things environment, user behavior has become more complex. The Spatio-temporal behavior, traffic behavior and demand behavior of users accessing resources are major factors affecting system security [22]. For example, traditional security mechanisms such as identity authentication and access control can only solve the problem of users' static management. Still, they can not solve the problem of users' behavior trust. When users access resources, they can obtain legal login access, but the behavior may not be trusted. For example, when a user frequently accesses a device's data, it can indicate that the user has a certain risk. Determining the user's trust level according to the user's historical behavior is a problem of dynamic trust management. The research on dynamic trust management technology is of great practical significance to ensure the reliable operation of the border protection system, safe sharing and trusted utilization of resources [19].

The research on user behavior trust mainly includes the trust evaluation of user's historical behavior, the prediction of future behavior trust, and the monitoring of real-time behavior trust, in which user behavior trust evaluation is the basis and user behavior trust control is the purpose. Because the evaluation of user behavior trust is

based on the behavior evidence of past communication, and it is a dynamic form of trust, therefore, Judging future behavior based on users' historical behavior is more challenging [14]. Evaluating the credibility of user behavior can play a role in finding suspicious users as soon as possible. Therefore, in evaluating the credibility of user behavior, we introduce a trust evaluation method based on the Transformer network. Through the trust evaluation method based on the Transformer model, the user behavior credibility is calculated, and the trust level is defined. The user's access rights can be adjusted in real-time according to the trust level to protect the system's security.

2 Related Work

Many scholars at home and abroad have done a lot of research on user behavior trust evaluation and established a variety of trust evaluation models. In 2010, Tian *et al.* Mainly discussed the importance and evaluation strategies of user behavior trust evaluation in cloud computing, including trust object analysis, principles for evaluating user behavior trust, basic ideas for evaluating user behavior trust, behavior trust evaluation strategy for each visit, behavior trust evaluation strategy for long access, etc. It lays the theoretical foundation of trust for the practical application of cloud computing [13]. In 2011, Yang *et al.* Proposed a statistical user behavior trust evaluation algorithm based on a cloud model. The algorithm uses a one-dimensional normal cloud model to extract the threshold of each behavior type and the membership degree of each user relative to the threshold. It then combines the membership degree and behavior weight to calculate the user's behavior trust evaluation value. Behavior trust is the basis of intradomain trust and recommendation trust; it is helpful for users to further dynamically authorize access control [20].

In 2013, Wu *et al.* proposed a cloud computing trust evaluation model based on D-S evidence theory and sliding window. Based on interactive evidence, the D-S evidence theory is used to calculate the direct trust of entities. Through the improved fusion method, the conflict of recommendation trust as second-hand evidence can be eliminated as far as possible. Finally, combined with recommendation trust, it reveals the credibility of entitie [17]. In 2014, Singh *et al.* proposed a trust evaluation mechanism, which calculates the user's absolute trust in the service provider based on the user's previous experience with the service provider and the recommendation of friends and third parties [12].

In 2015, Hosseini *et al.* proposed a user behavior trust evaluation method considering user identification code and MAC address. This method includes the user's new score, the behavior score of the past score, the score of other entities and the score of other cloud computing providers. The principle of calculating the score is given, and the total confidence of the end-user is obtained in the form of a score. This method helps to identify malicious users and negative behaviors [6]. In the same year, Li *et al.* proposed a comprehensive algorithm combining the entropy method and analytic hierarchy process. The evaluation becomes a dynamic model by using the entropy method to reflect the pattern of the basic law of user behavior evidence. It weakens the subjectivity of the simple use of the analytic hierarchy process. This method can effectively distinguish the abnormal behavior of users [9].

In 2016, Rashid *et al.* studied detecting such insiders by using a new method of hidden Markov model to detect abnormalities in the behavior, which may indicate an attack and provide a useful method to solve the challenge of internal threats [11]. In 2019, Yang *et al.* established a general access control game model that can reflect both trust and risk, which solved the prisoner's dilemma in the traditional access control game model without user behavior trust [16].

In 2021, Wu *et al.* Combined the analytic hierarchy process with fuzzy Petri-net theory and proposed a user behavior trust evaluation method based on Fuzzy Petri-net. In the trusted network, it can effectively evaluate the good trust of users in the network [18].

To sum up, we roughly divide the relevant work into three categories. These three categories have different concerns, but the three categories also have different limitations. For example, for traditional security methods such as access control and authorization, it is considered that internal users are always trusted, so the trust of user behavior is ignored. For the trust evaluation method based on non-learning, the evaluation process needs a series of predefined trust attributes, which are not comprehensive and subjective. There are also limitations to the current learning-based approach. Firstly, the trust evaluation time of algorithms used in these methods is long, occupying resources and increasing costs. Second, the algorithms used in these methods are not accurate enough to evaluate user behavior, directly affecting the evaluation results. Third, at present, in the evaluation of user behavior trust, people often evaluate directly according to the results generated by user behavior and rarely investigate the behavior evidence in the process of user behavior, thus ignoring the direct analysis of historical user behavior and the reliability, rationality and trust sharing of the results are limited [2, 3, 10].

In order to overcome these limitations, we propose to use the Transformer network with unsupervised pre-training to evaluate user behavior, which can reflect the long-term situation of user behavior and avoid subjectivity in the evaluation process. It will not cause errors due to the change of user behavior evidence value at a certain time and give full play to the role of historical data value. At the same time, the parallelism of transformer network will save us some training time and reduce the time cost. Moreover, the added unsupervised pre-training stage can improve the generalization ability of the trust model and improve the accuracy of trust evaluation.

3 Dynamic Trust Evaluation Method of User Behavior Based on Transformer

Due to the diversified, heterogeneous and massive characteristics of Internet of things terminal devices, before establishing the trust evaluation model, we first analyze the behavior of power end users. At this stage, the behavior of end-users is divided into normal user behavior, abnormal user behavior and malicious user behavior. Then we propose our trust evaluation model. Finally, the data set is divided into a training set and test set to train and test our trust evaluation model to predict the future behavior of end users and perform trust evaluation based on the similarity between the predicted behavior sequence and the actual real behavior sequence.

3.1 Analysis of User Behavior

With the gradual expansion of the scale of users in the Internet of things environment, user behavior has become more complex, which will also bring security risks. In this paper, the behavior of users in the Internet of things environment is divided into three categories: 1. Normal user behavior; 2. Abnormal user behavior: abnormal user behavior mainly refers to the fact that some attributes of power Internet of things users are very different from common attributes, such as login location, access resources, history, *et al.*; 3. Malicious user behavior. See Table 1 for details.

There are a large number of end-users, and their behaviors are diverse. There are many types and states. To facilitate analysis, we set the total number of end-user behaviors to m, and then the user's behaviors can be expressed as the following sets in timestamp i:

$$UB=\{\mathrm{ub}_{ij}|\mathrm{i},\mathrm{j}\in \mathrm{N}, 0<\mathrm{i}\leq t, 0<j\leq m\} \quad (1)$$

where, ub_{ij} indicates the j behavior of the end-user in the timestamp i.Then the end-user behavior value within t timestamps can be expressed in a matrix as:

$$UB=\begin{Bmatrix} ub_{11} & ub_{12} & \cdots & ub_{1m} \\ ub_{21} & ub_{22} & \cdots & ub_{2m} \\ \vdots & \vdots & \cdots & \vdots \\ ub_{t1} & ub_{t2} & \cdots & ub_{tm} \end{Bmatrix} \quad (2)$$

where, ub_{tm} indicates the m behavior of the end-user within the timestamp t.

3.2 Design of Transformer Network

The dynamic trust evaluation model based on the Transformer network needs to complete the design of the Transformer network before trust evaluation.

The traditional Transformer was proposed by Vaswani *et al.* [15]. The traditional CNN and RNN are abandoned in the Transformer, and the whole network structure is completely composed of an attention mechanism.

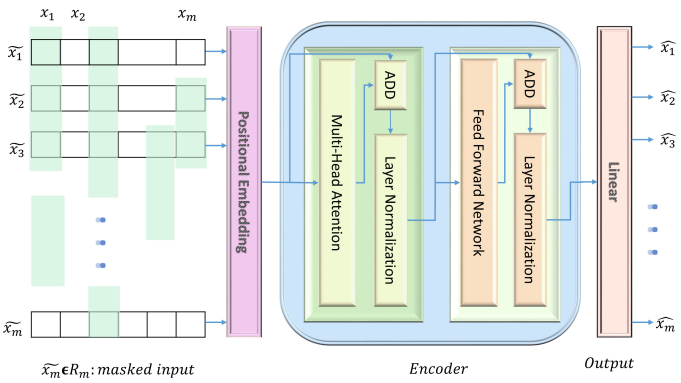


Figure 1: Detailed structure of unsupervised pre-training Transformer network

This makes Transformer have great advantages in parallel computing and long-distance dependent learning. Parallel computing is conducive to improving the training speed when the data set is large. However, the traditional Transformer structure can not be fully applicable to this field, especially the softmax layer in the output module, which is usually related to classification and probability. Therefore, the Transformer structure adopted in this paper only retains the input module, encoder module and output module without the softmax layer. This design is because the decoder module is suitable for the translation in NLP without specifying the length of the output sequence in advance, and the decoder module needs the real output sequence as input, so it is not suitable for the task of this paper. The most important thing is that only using the encoder allows us to use about half of the model parameters, which will benefit calculation and learning (for example, avoiding overfitting). At the same time, we use the input mask to mask the future information and pre-train the model [21] to improve the model's generalization ability. The specific network structure is shown in Figure 1.

Input module. This module includes position coding. Since there is no cycle or convolution in the model, position coding is added before the data is input into the encoder to use the sequence information of the sequence. The main functions used are as follows:

$$PE_{(pos,2i)} = sin(pos/10000^{2i/d_{model}}) \quad (3)$$

$$PE_{(pos,2i+1)} = \cos(pos/10000^{2i/d_{model}}) \quad (4)$$

where, d_{model} is the number of input features, pos indicates the current location, $PE_{(pos,2i)}$ indicates the location information when the dimension is even and $PE_{(pos,2i+1)}$ represents the location information when the dimension is odd.

Encoder. The feature extraction process used to specify the input is composed of N encoder layers stacked. Each encoder layer is divided into two sub-layers. Each sub-layer includes two parts. The first sub-layer includes a multi-head attention layer, residual connection, and

Table 1: User behavior analysis

Normal User Behavior	Abnormal User Behavior	Malicious User Behavior
User authentication (legal IP)	User authentication (illegal IP)	SQL Injection
Download behavior (normal)	Download behavior (abnormal)	Port Scan
Access resource type (within permission)	Access resource type (unauthorized access)	IP Spoofing
Number of resource usage	Number of resource usage	DDOS
Retrieval behavior	Retrieval behavior	SYNFlood
Download behavior	Download behavior	Replay Attacks
		Network Monitoring Attacks
		Virus Attacks

layer normalization. The second sub-layer includes a fully connected feedforward network, residual connection, and layer normalization, as shown in Figure 1.

Self-attention mechanism: The self-attention mechanism in the encoder makes Transformer have advantages in long-distance dependent learning. The attention calculation rules are as follows:

$$Attention(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V \tag{5}$$

where, Q, K, V it is a matrix obtained by linear transformation through the input of the self-attention layer, which is the query, key and value of attention, respectively, d_k is the number of columns of Q, K matrices, that is, the vector dimension, and softmax is a normalized exponential function, $\sqrt{d_k}$ is the numerical scaling to avoid sharp distribution.

Multi-head attention mechanism: The data will first pass through a multi head attention layer (composed of multiple self attention layers) in the encoder. This structure allows each attention mechanism to optimize different feature parts to balance the possible deviation caused by the same attention mechanism. The calculation method of multi-head attention is as follows:

$$MultiHead(Q, K, V) = Concat(head_1, \cdots, head_h)W^O$$

where, $Concat$ represents the splicing of vectors, W^O is the weight matrix, $head_i$ from the self-attention layer, it is:

$$head_i = Attention(QW_i^Q, KW_i^K, VW_i^V)$$

Fully connected feedforward network: after getting $MultiHead(Q, K, V)$, it will be sent to the fully connected feedforward network of the encoder, which takes out the average attention value and converts them into a form that is easier to deal with in the next layer. The feedforward network has two layers. The activation function of the first layer is, and the second layer is a linear activation function. The operation function of the feedforward network can be expressed as:

$$FFN(Z) = \max(0, ZW_1 + b_1)W_2 + b_2$$

where, Z is the output of the multi-head attention layer, which is also the input of the feedforward network, W_1, W_2 is the weight matrix and b_2 is the bias term.

Residual connection layer and layer normalization: with the increase in the number of network layers, the parameters may start to be too large or too small after multi-layer calculation, leading to an abnormal learning process and slow convergence of the model. Therefore, a certain number of layers will be followed by the normalization layer for numerical normalization to make its characteristic values within a reasonable range.

Output module. The final data is transferred to the output processing layer with a linear layer. The linear layer can flatten the attention value from the network.

Unsupervised pre-training. Usually, the whole output result will be embedded at one time during training, but in theory, the output of the decoder will not produce the final result at one time. Still, it will be synthesized from the last result again and again. Therefore, future information may be used in advance. Therefore, we use the input mask to mask the future information to pre-train the model, as shown in Figure 1. Unsupervised pre-training through input mask can use but not even use additional unlabeled data; that is, the generalization ability of the model can be improved by reusing existing data samples.

3.3 Dynamic Trust Evaluation Model of User Behavior Based on Transformer Network

In order to avoid subjectivity in the evaluation process, the model will not generate errors due to changes in the user's behavioral evidence value at a certain moment, and give full play to the role of historical data values. At the same time, user behavior data in the IoT environment is a typical collection of time series data, the tansformer network can not only capture thr sequence characteristics of user behavior, but also it has great advantages in computing and long-distance dependent learning, and parallelized computing is beneficial to improve the training speed in the case of large data sets. Therefore, according to the user behavior matrix generated in Section 2.1, we will use

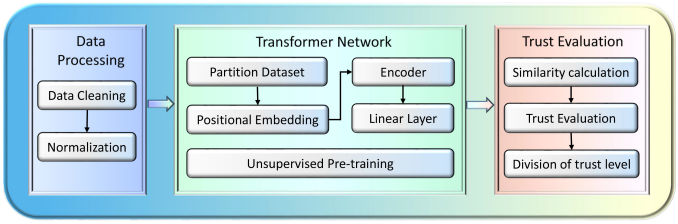


Figure 2: User behavior trust evaluation model

the constructed Transformer network to establish a user behavior dynamic trust evaluation model. The structure of the trust evaluation model is shown in Figure 2.

The trust evaluation model is mainly divided into three parts: data processing part, network training part and trust evaluation part. The data processing part is to process the original data as the input data of the Transformer network in the next stage. In the part of network training, the unsupervised pre-training of the Transformer network is carried out first. Then the feature extraction is carried out according to the historical input data through the feature extraction module of the Transformer network. At the same time, the Transformer network is continuously optimized according to the extraction results and optimizer. The trust evaluation part inputs the test set into the trained network for prediction. By calculating the similarity between the predicted behavior and the actual behavior, the trust level will evaluate the user's credibility to detect the user's trust state dynamically.

After the Transformer network is constructed according to Section 2.2, the training shall be started until the generalization ability of the model is not changing or the number of training iterations is reached. Then the trained network shall be used to evaluate the trust in user behavior. The specific steps are as follows.

Step 1. Preprocess the original data set, including data cleaning and normalization. Data cleaning includes deleting outliers and supplementing missing values in the data set. Data normalization can accelerate the convergence speed of gradient descent and improve the performance of the prediction model. Due to the different orders of magnitude of different user data, if used directly without processing, the difference of orders of magnitude will lead to the dominance of attributes with large orders of magnitude, contrary to the actual situation. Therefore, it is necessary to normalize the data to convert the data of different specifications into a unified specification so that the data can be normalized between 0 and 1 . The formula is as follows:

$$\delta_{ij} = \frac{ub_{ij} - ub_{\min}}{ub_{\max} - ub_{\min}} \tag{6}$$

where, ub_{ij} represents the actual value of user behavior, $ub_{\max}$ and $ub_{\min}$ represent the maximum and minimum values of user behavior, respectively. The user behavior matrix in Section 2.1 can be expressed as:

$$UB = \begin{Bmatrix} \delta_{11} & \delta_{12} & \cdots & \delta_{1m} \\ \delta_{21} & \delta_{22} & \cdots & \delta_{2m} \\ \vdots & \vdots & \cdots & \vdots \\ \delta_{t1} & \delta_{t2} & \cdots & \delta_{tm} \end{Bmatrix} \tag{7}$$

Step 2. Divide the processed data set into a training set and test set.

Step 3. Firstly, the divided training set is used as the input of the Transformer network, and the Transformer network is unsupervised pre-trained. By setting some input data values to 0, the model is used to predict these data. Then, we mask each input variable sequence with the proportion r , so that on each variable, the period with the average length of l_m is shielded, and the average length l_u of the unshielded section after each period is $\frac{1-r}{r} l_m$, the model attempts to predict the complete and undamaged input vector x_i at each time step. Selecting this masking mode makes the model learn to focus on the previous and subsequent parts of a single variable and the existing values of other variables in the time series, so you can learn to model the interdependence between variables. Finally, save the Transformer network after pre-training.

Step 4. Send the test set to the saved Transformer network and fine-tune the network. First, compare the results with the previous test results until the generalization ability does not change or the number of training cycles is reached. Then save the fine-tuned network. Finally, the test set is input into the final saved Transformer network to predict the user behavior.

Step 5. According to the prediction results of the trust evaluation model, the formula is applied to calculate the similarity $Sim(UB_R, UB_P)$ between the predicted behavior and the actual behavior. Since each action is encoded as a one-hot vector, the similarity distance between vectors is used to calculate the similarity for each action. The calculation formula is as follows:

$$Sim_i = 1 - \frac{(UB_i^R - \overline{UB_i^R}) \bullet (UB_i^P - \overline{UB_i^P})}{\left\| UB_i^R - \overline{UB_i^R} \right\|_2 \left\| UB_i^P - \overline{UB_i^P} \right\|_2} \tag{8}$$

Therefore, the average similarity calculation formula of behavior sequence is as follows:

$$Sim(UB_R, UB_P) = \frac{1}{x} \sum_{i=1}^{x} Sim_i(UB_i^R, UB_i^P) \tag{9}$$

where, $Sim(UB_R, UB_P) \in [0, 1]$, UB_R, UB_P represent the user's actual action and prediction action respectively, $\overline{UB_R}$ and $\overline{UB_P}$ are the mean values of

UB_R and UB_P respectively, and $X \bullet Y$ represents the dot product of vector X and vector Y . The smaller value of the $Sim(UB_R, UB_P)$, the greater the similarity between prediction behavior and actual behavior.

Step 6. Compare the similarity value calculated in step 5 with the trust threshold in Table 2, evaluate the trust of user behavior and classify the trust level.

The user behavior trust evaluation algorithm based on Transformer is as Algorithm 1.

Algorithm 1 Dynamic trust evaluation algorithm of user behavior based on Transformer network

Input: Enter the user behavior set UB , the maximun number of iterations is M , and divide the user behavior set into training set UB_t and test set UB_v.
Output: The trust level of the user corresponding to the user behavior.
for $m = 1 \rightarrow M$ //m represents the number of iterations **do**
  if $m \leq M$ **then**
    Training Transformer network. //The training process includes unsupervised pre-training of Transformer network
  else
    break //Save the trained Transformer network and exit the loop.
  end if
end for
while true **do**
  function *Transformer(to predict)* //The trained Transformer network is used to predict the action sequence
  predictAction = Transformer(to predict)
  return *predictAction*
  end function
end while
Calculate the similarity between *predictAction* and *realaction* and compare it with the trust threshold to divide the trust level.
return The trust level of the user corresponding to the user behavior. //Output results

4 Experiment and Result Analysis

In this section, we will describe the experimental environment and super parameters of the model in detail, and verify the effectiveness of our model on a public data set. In addition, we will conduct comparative experiments with different models to prove the performance of our model.

4.1 Experimental Environment and Data Preparation

This experiment uses Python 3.6 as the model programming environment, Pycharm Professional 2020.1 as the programming platform and GPU for acceleration. The GPU model is NVIDIA TITAN XP, the operating system is Win10, and the version is 20H2. The Transformer algorithm used in this paper is implemented by Pytorch.

This experiment uses the CERT-IT (Insider Threat) data set [1] constructed from the internal threat center of Carnegie Mellon University. This data set is the data collected from the real enterprise environment, simulate the three main attack behavior data of system destruction, information theft and internal fraud implemented by insiders, and a large number of normal background data, and records the user behavior in CSV format. It includes file access rights, various attributes of files, addition, deletion and modification of files by users, Email sending and receiving, use records of hardware devices such as mobile storage devices and printers, HTTP access and system login, jobs and departments, etc. The CERT dataset provides comprehensive behavior observation data to describe the user behavior model.

We extracted about 20000 normal behaviors from the normal data background and about 5000 behaviors during the attack. These behaviors are regarded as the untrusted behaviors of users. The extracted normal background data is used to train the Transformer model to determine the best super parameters of the model and establish the corresponding trust model for each user. The evaluation status of the trust model is tested by the data extracted during the attack, that is, the untrusted behavior of the user.

4.2 Model Evaluation Index

When evaluating the dynamic trust evaluation model based on the Transformer network, on the one hand, in order to directly calculate the prediction error of the overall task, the accuracy of the predicted value of user behavior is adopted. The calculation formula is as follows:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{10}$$

where, true positive (TP) indicates that the real category of the sample is positive, and the result predicted by the model is also positive; false positive (FP) indicates that the real category of the sample is negative, but the model incorrectly predicts it as positive; true negative (TN) indicates that the real category of the sample is negative, and the model correctly predicts it as negative; false negative (FN) indicates that the real category of the sample is positive, but the model incorrectly predicts it as negative.

On the other hand, by comparing the training time of different algorithms, that is, the time required to train a batch of samples, we can see the advantages of Transformer network parallelism.

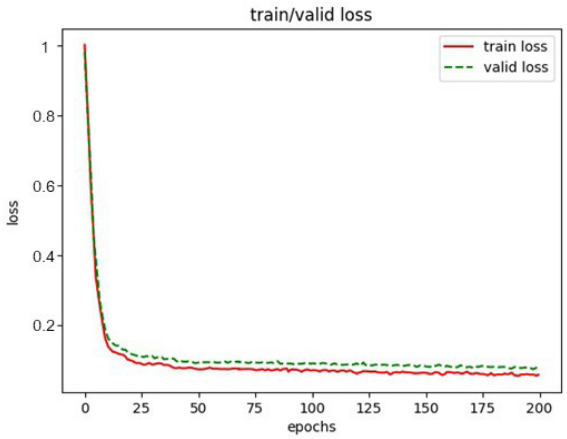


Figure 3: Loss function diagram of the model

4.3 Level of Trust Assessment

By analyzing the user's historical behavior, using the trust evaluation model to evaluate their behavior trust, establishing the user behavior trust evaluation level, and dynamically assigning service levels to them according to the user's trust level, so as to realize the security supervision and control of the user's entire access process. At the same time, by monitoring the user's trust status, the access of abnormal and malicious users can be blocked in time. Therefore, this paper divides the user trust level into 4 levels. The trust level from high to low is: trusted, generally trusted, generally untrusted and untrusted. The corresponding value range is shown in Table 2.

When the user behavior evaluation result is within the general untrusted range, mark the user as an abnormal user; mark the user as a malicious user when the user behavior evaluation result is not credible; When the evaluation result is within the credible and general credible range, mark the user as a normal user and respond to the attribute assignment requirement.

4.4 Experimental Results and Analysis

The super parameters d_k (query and key dimension), d_v (value dimension), d_{model} (generation and output dimension of all sublayers and embedded layers in the model), h (number of attention headers) and d_{ff} (dimension of feedforward network layer) of the encoder in the Transformer network used in this experiment are 16, 16, 128, 8 and 256, respectively. During pre-training, the batch_size is 32, and the training iteration is 700 times. Adam [7] is used as the optimizer, and the learning rate is 0.001. The decline factor $drop_\alpha$ of learning rate is 0.9. When fine-tuning the model, batch_size is 50, 200 training iterations are performed, and the Adam optimizer parameters remain unchanged. Figure 3 shows the loss function curve during model training and verification, which shows that our model can fit well.

In order to verify the effectiveness of our model, using the trained Transformer network, we conducted experiments on untrusted behavior to evaluate the effectiveness of the proposed method. We choose the last 100 actions in the normal state as the current action to predict the next 20 actions and then calculate the similarity value between each action in the predicted sequence and each action in the real sequence in the attack state. For comparison, we also predicted some sequences under normal conditions, and the results are shown in Table 3.

It can be seen from Table 3 that when there is a normal behavior sequence, the similarity is close to 0. According to Table 2, it is divided into trust level and response authority allocation. From the last row of Table 3, it can be seen that when there is a malicious behavior sequence, the similarity is almost close to 1. According to the trust evaluation table, it is divided into untrusted levels, and the abnormal behavior sequence can also be accurately divided into general untrusted levels that do not respond to its permission assignment. It can be clearly seen that the trust levels of normal sequence (the first and second lines), abnormal sequence (the third and fourth lines) and malicious sequence (the last line) are significantly different and can accurately divide the trust level, which verifies that the similarity is useful for evaluating user behavior.

In order to verify the performance of our model, we will conduct comparative experiments with the following algorithms.

LSTM [5]: a commonly used time series modeling and prediction model, which can learn and bridge the minimum time lag of more than 1000 discrete time steps through constant error rotation and forced constant error flow in special units and solve the complex and artificial long time lag task never solved by the previous cyclic network algorithm.

LSTNets [8]: a novel deep learning framework uses a convolutional neural network (CNN) and cyclic neural network (RNN) to extract short-term local dependence patterns between variables and find long-term patterns of time series trends. It can effectively solve the problem of multivariate time series prediction.

Transformer [15]: Transformer abandons the common time-series and convolution-based modeling mechanism, which is also a more simple and predictable parallel model.

We select 20000 normal behaviors to be dynamically generated into several mini-batches with a size of 32 and train the model 200 times, respectively. The d_{model} parameter of Transformer without pre-training is set to 157. The experimental results of the four algorithms are shown in Figure 4 and Figuer 5. In terms of training time, it can be clearly seen from Figure 4 that the Transformer network has the best performance, with a time of 7.55s, followed by our method, with a time of 15.67s, because the Transformer network we use an unsupervised pre-training process, which will increase a certain time cost. In terms of the accuracy rate of the model, it can be concluded from Figure 5 that the best performance is when we add

Table 2: Trust level division of user behavior

Confidence Threshold	Trust Level
[0, 0.25]	Trusted
(0.25, 0.5]	Generally trusted
(0.5, 0.75]	Generally untrusted
(0.75, 1]	Untrusted

Table 3: Similarity between predicted sequence and real sequence

Behavior Sequence	Similarity	Trust Level	Grant Permissions
Normal sequence 1	0.04	Trusted	Y
Normal sequence 2	0.33	Generally trusted	Y
Abnormal sequence 1	0.71	Generally untrusted	N
Abnormal sequence 2	0.68	Generally untrusted	N
Malicious sequence	0.98	Untrusted	N

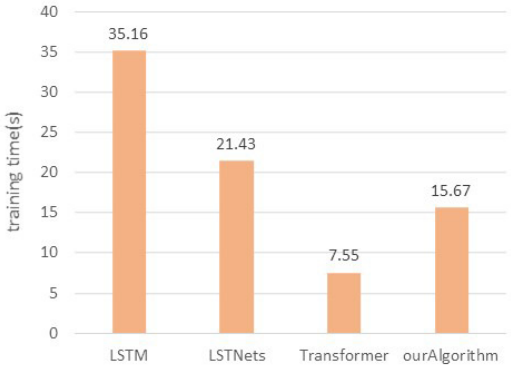


Figure 4: Training time of different trust evaluation methods

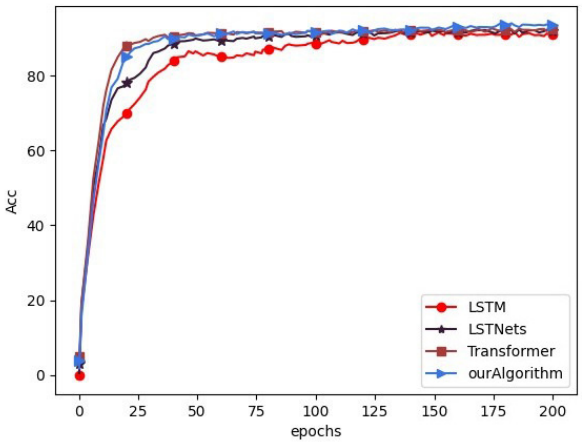


Figure 5: Accuracy of trust evaluation of different algorithms

the Transformer network with pre-training, and the accuracy rate reaches 93.481%, followed by the Transformer network, reaching 92.304%. The accuracy rates of LSTM and LSTNets are relatively low, 91.052% and 91.885%, respectively.

Based on the above two aspects, for different networks, the Transformer network with unsupervised pre-training is better than the other two networks in terms of training time and accuracy; for the same network, the accuracy of the transformer network with unsupervised pre-training is increased by 1.177% at the cost of a certain time cost. Therefore, compared with LSTM, LSTNets and Transformer, the method proposed in this paper is more suitable for user behavior trust evaluation.

5 Conclusion

This paper proposes a dynamic trust evaluation method of user behavior based on Transformer. This method first analyzes the historical behavior data of power end users, normalizes the data, then carries out unsupervised pre-training on the Transformer network used, models the user's behavior pattern, and establishes the trust model of the user's historical behavior. Finally, the established trust model predicts the future behavior of a user, calculates the similarity between the predicted behavior and the user's actual behavior, and compares it with the credibility threshold to realize the dynamic trust evaluation of user behavior. Comparative experiments show that using the Transformer network for trust evaluation can not only save evaluation time, also improve the accuracy of prediction, but also increase the unsupervised pre-trained Transformer network at the cost of time cost, which can improve the accuracy of prediction and the generalization ability of the model, It shows that the dynamic trust evaluation method based on Transformer network is better than other models, and can be well applied in the field of user behavior trust evaluation.

Acknowledgments

This work was supported in part by the State Grid Jiangxi Information & Telecommunication Company Project "Research on de-boundary security protection technology based on zero trust framework" under Grant 52183520007V.

References

[1] Carnegie Mellon University Software Engineering Institute[EB/OL]. *CERT-IT Data Set*.

[2] Y. H. Gong and L. Chen, "Trust evaluation of user behavior based on entropy weight method," in *International Conference on Computer Engineering and Networks*, pp. 670–675. Springer, 2020.

[3] L. S. Han and X. J. Yu, "Research on cloud end-user behavior trust evaluation model based on sliding window," in *2021 IEEE Conference on Telecommunications, Optics and Computer Science (TOCS)*, pp. 270–277. IEEE, 2021.

[4] Z. G. He, "Multi-parameter and time series based trust for iot smart sensors.," *International Journal of Network Security*, vol. 22, no. 4, pp. 589–596, 2020.

[5] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[6] S. B. Hosseini, A. Shojaee, and N. Agheli, "A new method for evaluating cloud computing user behavior trust," in *2015 7th Conference on Information and Knowledge Technology (IKT)*, pp. 1–6. IEEE, 2015.

[7] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[8] G. Lai, W. C. Chang, Y. M. Yang, and H. X. Liu, "Modeling long-and short-term temporal patterns with deep neural networks," in *The 41st international ACM SIGIR conference on research & development in information retrieval*, pp. 95–104, 2018.

[9] J. J. Li and L. Q. Tian, "User's behavior trust evaluate algorithm based on cloud model," in *2015 Fifth International Conference on Instrumentation and Measurement, Computer, Communication and Control (IMCCC)*, pp. 556–561. IEEE, 2015.

[10] A. Mansouri and M. S. Bouhlel, "Trust in ad hoc networks: A new model based on clustering algorithm.," *International Journal of Network Security*, vol. 21, no. 3, pp. 483–493, 2019.

[11] T. Rashid, I. Agrafiotis, and J. R. Nurse, "A new take on detecting insider threats: exploring the use of hidden markov models," in *Proceedings of the 8th ACM CCS International workshop on managing insider security threats*, pp. 47–56, 2016.

[12] S. Singh and D. Chand, "Trust evaluation in cloud based on friends and third party's recommendations," in *2014 Recent Advances in Engineering and Computational Sciences (RAECS)*, pp. 1–6. IEEE, 2014.

[13] L. Q. Tian, C. Lin, and Y. Ni, "Evaluation of user behavior trust in cloud computing," in *2010 International Conference on Computer Application and System Modeling (ICCASM 2010)*, vol. 7, pp. V7–567. IEEE, 2010.

[14] Y. Tian, "Research on privacy protection and user behavior trust based on attribute encryption," *Taiyuan University of Technology*, 2018.

[15] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[16] Y. Wang, L. Q. Tian, and Z. G. Chen, "Game analysis of access control based on user behavior trust," *Information*, vol. 10, no. 4, p. 132, 2019.

[17] X. Wu, R. Zhang, B. Zeng, and S. Zhou, "A trust evaluation model for cloud computing," *Procedia Computer Science*, vol. 17, pp. 1170–1177, 2013.

[18] Z. A. Wu, L. Q. Tian, Z. G. Wang, and Y. Wang, "Web user behavior trust evaluation model based on fuzzy petri net," in *2021 IEEE 6th International Conference on Big Data Analytics (ICBDA)*, pp. 344–348. IEEE, 2021.

[19] P. S. Xie, X. Q. Wang, X. J. Pan, Y. F. Wang, T. Feng, and Y. Yan, "Blockchain-based trust evaluation mechanism for internet of vehicles nodes," *International Journal of Network Security*, vol. 23, no. 6, pp. 1065–1073, 2021.

[20] X. Yang, L. Liu, and R. Zou, "A statistical user-behavior trust evaluation algorithm based on cloud model," in *2011 6th International Conference on Computer Sciences and Convergence Information Technology (ICCIT)*, pp. 598–603. IEEE, 2011.

[21] G. Zerveas, S. Jayaraman, D. Patel, A. Bhamidipaty, and C. Eickhoff, "A transformer-based framework for multivariate time series representation learning," in *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pp. 2114–2124, 2021.

[22] J. Zhao, P. Guo, H. Z. Deng, J. Wu, Y. J. Tan, and M. Zhang, "Influence of statistical characteristics of user behavior on performance and reliability of communication network," *Journal of Communication*, vol. 34, no. 1, pp. 43–50, 2013.

Biography

Xiu-wen Yu biography. She was born in Jan. 1995. She is a master student at North China Electric Power University. Her major research field is information security and privacy protection. E-mail: yuxiuwen0129@163.com.

Rong Huang biography. She was born in 1995. She is a master student at North China Electric Power University. Her major research field is information security and privacy protection. E-mail: RongHuang@ncepu.edu.cn.

Yuan-cheng Li biography. He was born in Aug. 1972. He is a professor and a supervisor of Doctoral student at North China Electric Power University. His major research field is information security and privacy protection, cryptography and blockchain, artificial intelligence and security. E-mail: ncepua@163.com.

Ri-xuan Qiu biography. He works for the Information communication Branch of State Grid Jiangxi Electric Power Co., LTD. E-mail: qiurixuanwork@163.com.

Xin Zhou biography. He works for the Information communication Branch of State Grid Jiangxi Electric Power Co., LTD. E-mail: xzh1882@163.com.

Liang Liang biography. He works for the Information communication Branch of State Grid Jiangxi Electric Power Co., LTD. E-mail: l_liang@foxmail.com.

Si-tong Jing biography. He works for the PowerChina Jiangxi Electric Power Engineering Co., LTD. E-mail: aurrucy@126.com.

A Data Sharing Scheme in NDN Based on MOR Primitive

Jiaoli Shi[1,2], Anyuan Deng[1,2], Chao Luo[1,2], Shimao Yao[1,2], and Kai He[3,4]
(Corresponding author: Shimao Yao)
School of Computer and Big Data Science, Jiujiang University, Jiujiang, Jiangxi, China[1]
Email: 407081693@qq.com
Institute of Information Security, Jiujiang University, Jiujiang, Jiangxi, China[2]
School of Computer Science and Artificial Intelligence, Wuhan Textile University, Wuhan, Hubei, China[3]
Hubei Clothing Information Engineering Technology Research Center, Wuhan, Hubei, China[4]
(Received Apr. 27, 2022; Revised and Accepted Oct. 12, 2022; First Online Oct. 15, 2022)

Abstract

This work is aimed at a data sharing scheme over NDN (Named Data Networking) with multiple demands: loose-couple, fine-grained control, and dynamic authority management. Firstly, a data access control scheme is proposed. Publisher generates data, encrypts the data to ciphertext, binds the ciphertext with an access policy, and publishes the ciphertext on an NDN router. Subscriber can access the part of data authorized when his private key meets the policy bound to the ciphertext, which realizes the loose-couple and fine-grained access control. The scheme can be constructed through Matrix computation, based on the MOR primitive instantiated by LWE. Secondly, a dynamic authority management method is presented. A subscriber's latest identity public key is generated and stored on edge routers. When the subscriber applies to ciphertext, the edge routers attach the latest identity public key of the subscribers to the ciphertext and send it to the subscriber. This design can drive a subscriber to use his latest private key.

Keywords: Attribute Revocation; Attribute-Based Encryption; Data Sharing; Dynamic Authority Management

1 Introduction

More than half of the Internet traffic, such as on YouTube, is carried by ICN (Information-Centric Network) [11]. NDN (Named Data Networking), as a specific ICN, has been regarded as the most potential network architecture. NDN architecture is data-centered. After consumers publish an interest packet, any router can find matched content in its content store and send it to consumers along the reverse path of the interest packet. It is stated that security should be an attribute of packets in the NDN Protocol design principle [11].

However, Data secure sharing is challenging in an open sharing environment like NDN: 1) Loose coupling limits the order of authorization and encryption. If encryption is followed by authorization, the key needs to be defined in advance. However, in NDN, the provider does not know the exact identity information of the consumer, and the consumer does not know the exact identity information of the provider. The provider and consumer cannot transfer the key in advance, so authorization is required before encryption. 2) Fine-grained access control is both confidential and open. Confidentiality means that all unauthorized consumers should not be given access to the data; Openness means that authorized consumers have access to authorized portions of data.

Considering the above challenges, ABE (Attribute-Based Encryption) and IBE (Identity-Based Encryption) schemes can realize authorization before encryption, but IBE control granularity is not as good as ABE. In addition, a lot of key management is still required if the ID representing user permissions is bound to the ciphertext permission control block, such as the proxy re-encryption. It is not difficult to find that ABE autonomous access control mechanism is the most suitable data sharing method. On one hand, the attribute-based cryptography method of the mechanism realizes one-to-many broadcast encryption. The provider only needs to encrypt and publish data before being offline. Multiple consumers can download and decrypt the ciphertexts according to the private key issued by the key management agency anytime and anywhere. On the other hand, ABE mechanisms enable fine-grained control, and allows any number of consumers to share moving data of any length.

However, most of the cryptographic primitives used to construct ABE are based on bilinear pairings and cannot cope with the cryptographic crisis in the post-quantum era. Some researchers construct ABE schemes based on lattice cryptography under worst-case difficulty. In this paper, based on ABE and MOR (Many-to-one Recoding) primitives [4], an NDN data security access control

scheme is designed. This scheme has the characteristics of loose coupling, fine-grained control, flexible and portable key management, and uses version management to achieve dynamic permission reversion.

The main contributions of this paper can be summarized as follows:

1) We present an NDN data access control scheme based on LWE, in which, a) all unauthorized consumers cannot access data (confidentiality), and authorized consumers can access authorized data (openness); b) there is no coupling between a provider and a consumer in time, space, and synchronization.

2) We propose a consumer dynamic authority management method. On the premise of ensuring forward and backward security, the permission of a consumer can be updated directly and immediately.

2 Related Works

ABE scheme based on lattice theory is generally considered by scholars to have the characteristics of anti-quantum attack. In 2009, Regev [13] proved that the average difficulty of solving LWE problems was equivalent to the worst-case lattice problem.

In 2013, Gorbunov *et al.* [3] proposed ABE scheme based on TOR (Two-to-One Recoding) and instantiated TOR based on LWE. In 2014, Liu *et al.* [16] presented a lattice-based threshold hierarchical ABE scheme in the standard model. They divided attributes to different levels according to their important degrees without increasing the dimension of the lattices using a delegation mechanism. In 2019, Wang *et al.* [15] introduced Three-to-One Recoding and then constructed a ciphertext policy ABE scheme under the LWE assumption.

In 2019, Agrawal *et al.* [1] constructed a encryption scheme based on symmetric key attribute for NFA (Nondeterministic Finite Automata) under LWE assumption supporting unbounded length inputs. Their work extended the scope of application of the LWE-based ABE scheme. Tsabary [14] constructed a lattice-based CP-ABE scheme, in which an access policy was described as a form of t-CNF (Conjunctive Normal Form). t was a constant and a threshold, which denoted that each clause in a policy depended on at most t bits of the input.

Both Kim [8] and Datta *et al.* [2] supported multi-authority in their LWE-based ABE schemes. Kim [8] considered that most MA-ABE(multi-authority attribute-based encryption) schemes could only support predicates computed by monotone Boolean formulas because of their bilinear map foundation. They constructed a MA-ABE scheme supporting circuit predicates under LWE assumption, in which a user gets his structured key put together with multiple components issued by multiple authorities using FHE (fully homomorphic encryption). Datta *et al.* [2] also constructed a CP-ABE scheme under the LWE assumption. Their access policies were described by DNF formulas. Any party could become an authority and no central authority was required. But a bound s on policies was required and linear with the sizes of a public key, a secret key, and a ciphertext.

Up to now, no LWE-based ABE scheme has been found to support permission revocation, while ABE schemes based on bilinear pairs often use four types of methods to deal with the attribute revocation.

Type 1: Delay Revocation. That is, consumers or some of their permissions are revoked periodically, which leads to ciphertext re-encryption or key re-issue in batches. Such as in Li *et al.* [10].

Type 2: Narrow the Revoking Scope. the concept of consumer group or attribute group is introduced to narrow the scope of ciphertext re-encryption or key re-issue. Such as in Li *et al.* [6].

Type 3: Maintain Revocation Lists. Ciphertexts are associated with consumer attribute revocation lists. This method does not need to re-encrypt the ciphertext or re-issue the key but needs to maintain an attribute revocation list for each ciphertext. Such as in Xu *et al.* [5], Zhang *et al.* [7], He *et al.* [9] and Yan *et al.* [17]. In these works, when an attribute of a user was revoked, the non-revoked user could access the ciphertext without obtaining a new private key from AA.

Type 4: Set Timeliness of Attributes. The validity period or version number of attributes is set to control the timeliness of attributes, such as in Liu *et al.* [12].

3 Preliminary

3.1 LWE (Learning With Errors) Assumption

Gorbunov *et al.* [3] defined the (decisional) LWE problem Informally. For an integer $q{=}qn \geq 2$ from uniformly random numbers and a random error distribution $\mathbf{e} = \mathbf{e}(n)$ over $_q$, there is an negligible advantage to distinguish between the following pairs of distributions: $\{\mathbf{A}, \mathbf{A}^{\mathrm{T}}\mathbf{s}{+}\mathbf{e}\}$ and $\{\mathbf{A},\mathbf{u}\}$, wherein $\mathbf{A} \in {}_q^{n\times m}$, A^{T} denotes its transpose, $\mathbf{s} \in {}_q^n$, $\mathbf{e} \in {}_q^m$,$\mathbf{u} \in {}_q^m$.

3.2 MOR (Many-to-one Recoding) Primitive

Given a set of pseudo-random functions $Encode(\mathbf{.})$ and a recoding key rk, the following transformation can be performed. $(Encode(\mathbf{pk}_1,\omega),\cdots,Encode(\mathbf{pk}_{\mathrm{N}},\omega)) \mapsto Encode(\mathbf{pk}_{\mathrm{tgt}},\omega)$ wherein, the rk can be produced from a private key $\mathbf{sk}_b$ corresponding to any public key $\mathbf{pk}_b$, and ω denotes a uniformly Gaussian random seed.

3.3 Instantiation of MOR Primitive

According to Gorbunov *et al.* [3], TOR primitives was instantiated from LWE assumption as Formula (1):

$$\mathbf{R}^{\mathrm{T}}\begin{bmatrix}\mathbf{A}_0^{\mathrm{T}}\mathbf{s}+\mathbf{e}_0\\ \mathbf{A}_1^{\mathrm{T}}\mathbf{s}+\mathbf{e}_1\end{bmatrix}\approx\mathbf{A}_{\mathrm{tgt}}^{\mathrm{T}}\mathbf{s}, \tag{1}$$

wherein, $\mathbf{s}\in{}_q^n$ denotes a random number of module q. $\mathbf{R}\in{}_q^{2m\times m}$ denotes a low-norm matrix satisfied with $[\mathbf{A}_0||\mathbf{A}_1]\mathbf{R}=\mathbf{A}_{\mathrm{tgt}}$, and its transpose is $\mathbf{R}^{\mathrm{T}}$. The $\mathbf{R}$ maps into a recoding key rk. $\{\mathbf{A}_b\in{}_q^{n\times m}|b\in\{0,1\}\}$ maps into a public key $\{\mathbf{pk}_b|b\in\{0,1\}\}$. $\{\mathbf{T}_b\in{}_q^{m\times m}|b\in\{0,1\}\}$ maps into the private key $\{\mathbf{sk}_b|b\in\{0,1\}\}$. ${\mathbf{A}_b}^{\mathrm{T}}\mathbf{s}+\mathbf{e}_b$ maps into $Encode(\alpha_b,\omega)$. $\mathbf{e}_b$ satisfies the Gaussian distribution $D_{m,\sigma}$ and its order is $\sqrt{m}\cdot\sigma$. Based on LWE assumption, Formula (1) can be extended to Formula (2):

$$\mathbf{R}^{\mathrm{T}}\begin{bmatrix}\alpha_1^{\mathrm{T}}\mathbf{s}+\mathbf{e}_1\\ \mathbf{A}_2^{\mathrm{T}}\mathbf{s}+\mathbf{e}_2\\ \vdots\\ \mathbf{A}_{\mathrm{N}}^{\mathrm{T}}\mathbf{s}+\mathbf{e}_{\mathrm{N}}\end{bmatrix}\approx\mathbf{A}_{\mathrm{tgt}}^{\mathrm{T}}\mathbf{s}, \tag{2}$$

wherein, $\mathbf{R}\in{}_q^{Nm\times m}$.

3.4 Definition of MOR

Based on Formula (2), we define the MOR scheme as follows.

1) $Params(1^{\lambda\Lambda},d_{\max})$

The algorithm takes as input a security parameter λ and an upper bound $d_{\max}$ on the number of nested recoding operations, outputs global public parameters $PP=(n,X,q,m,s)$. Wherein, $n=n(\lambda)$ denotes the dimension of LWE. $X=X(\lambda)=D_{,\sqrt{n}}$ denotes an Error distribution. $B=B(n)=O(n)$ denotes an order of Error. $d_{\max}$ denotes the maximum number of times the algorithm $Recode(.)$ is run. That is the depth of $Recode(.)$. $q=q(n)=\tilde{O}(n^2d_{\max})^{d_{\max}}n$ denotes the module of LWE. $m=m(n)=O(n\log q)$ denotes the number of samples. $s=s(n)=O(\sqrt{n\log q})$ denotes a Gaussian parameter.

2) $Keygen(PP)$

The algorithm generates a matrix $\mathbf{A}$ and its trapdoor matrix $\mathbf{T}$, which will be act as a public key $\mathbf{pk}=\mathbf{A}$ and its private key $\mathbf{sk}=\mathbf{T}$. Wherein, $\mathbf{A}\in{}_q^{n\times m}$, $\mathbf{T}\in{}_q^{m\times m}$.

3) $Encode(\mathbf{pk},\omega)$

The algorithm selects two vectors: $\mathbf{e}\in X^m$ and $\mathbf{s}\in{}_q^n$, and outputs $\varphi=\mathbf{A}^{\mathrm{T}}\mathbf{s}+\mathbf{e}$.

4) $ReKeygen(\mathbf{pk}_1,\mathbf{pk}_2,\cdots,\mathbf{pk}_N,\mathbf{sk}_b,\mathbf{pk}_{\mathrm{tgt}})$

The algorithm selects $N-1$ discrete Gaussian matrices $\mathbf{R}_i\in(D_{,\sigma})^{m\times m}$, calculates $\mathbf{U}=\mathbf{pk}_{\mathrm{tgt}}-\sum_{i=1}^{n}\mathbf{A}_i\mathbf{R}_i$—$i\neq b$, calculates $\mathbf{R}_b=SampleD(\mathbf{A}_b,\mathbf{T}_b,\mathbf{U})$, and outputs the rk^{tgt}:

$$rk^{tgt}=\begin{bmatrix}\mathbf{R}_1\\ \mathbf{R}_2\\ \vdots\\ \mathbf{R}_N\end{bmatrix}\in{}^{Nm\times m}$$

5) $Recode(rk^{tgt},\varphi_1,\varphi_2,\cdots,\varphi_N)$ The algorithm calculates $\varphi_{\mathrm{tgt}}=\left[\varphi_1^{\mathrm{T}}||\varphi_2^{\mathrm{T}}|\cdots|\varphi_{\mathrm{N}}^{\mathrm{T}}\right]rk^{tgt}$.

6) $E(\varphi,\mu)$

The algorithm encrypts a plaintext μ: $\Upsilon=\varphi+\left\lceil\frac{q}{2}\right\rceil\mu(\mathrm{mod}q)$.

7) $D(\varphi',\Upsilon)$

The algorithm decrypts a ciphertext Υ: $\mu=(Round(\Upsilon_1-\varphi_1^{'}),\cdots,Round(\Upsilon_n-\varphi_n^{'}))$, wherein, $Round(x)$ is defined as follows.

$$Round(x)=\begin{cases}0, & \text{If}\,|x|<q/4.\\ 1, & \text{Otherwise.}\end{cases}$$

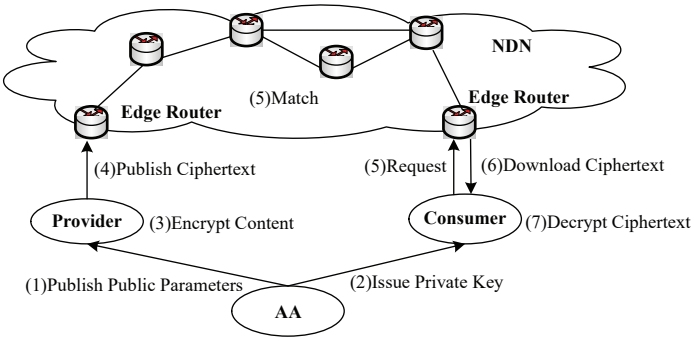


Figure 1: System model of NDN data sharing scheme

4 System Model

The NDN system model is shown as in Figure 1. It consists of providers, consumers, NDN routers and authorization centers. Providers publish contents anytime and anywhere; NDN is responsible for storing, matching and forwarding content. Consumers get contents anytime and anywhere. The authorization center is responsible for generating system public parameters and generating private keys for consumers.

Contents Confidentiality. Contents are encrypted before uploading and will not be accessible to NDN routers or unauthorized consumers.

Forward and Backward Security. Newly added consumers should be able to access previous encrypted

contents if their attributes meet access control policies. Consumers who have revoked one or more attributes cannot access that content if their remanent attributes do not meet the access control policy.

Collusion Resistance. Two legitimate consumers cannot combine their private keys to promote privileges they do not have.

5 Proposed Scheme

5.1 Framework

The framework of the proposed scheme is shown as Figure 2.

Initialization Phase. AA calls the *Initialization* algorithm to generate global public parameters PP. The *Setup* algorithm is invoked to generate attribute public keys $\{APK_i\}$ for all attributes, calculate the circuit output public key PK_{out} for different access control policies, and set the latest identity public key GPK_{sub} and an attribute private key USK_{sub} for each consumer. $\{APK_i\}$ and PK_{out} will be exposed to the provider, GPK_{sub} will be exposed to the semi-trusted edge routers, USK_{sub} will be sent to the consumer using the key exchange protocol. Any keywords of the content can be arbitrarily specified by AA in the form of attributes in the Setup algorithm for greater flexibility.

Data Preparation Phase. The provider invokes the *EncryptData* algorithm, encrypts the plaintext M with the symmetric key $K_{Content}$, and runs the symmetric encryption algorithm to obtain cm. The provider defines the access control policy *policy*, run the *BindingPolicy* algorithm for $K_{Content}$ using CP-ABE encryption to obtain cp, and upload $cm||cp$ onto the edge router.

Data Retrieval Phase. When the consumer applies for the content, the edge router runs the *AttachGPK* algorithm and sends $cm||cp_{attached}$ to the consumer, who uses its own attribute private key, runs the *MatchPolicy* algorithm, obtains the symmetric key $K_{Content}$, and then runs the *DecryptData* algorithm to read the plaintext.

Permission Adjustment Phase. AA runs the *UpdateKey* algorithm when a consumer's permission changes dynamically. The algorithm generates his latest identity public key GPK_{sub} and attribute private key USK_{sub}. The former is sent to the edge router and the latter to the consumer through the secret exchange protocol.

5.2 Construction of Our Scheme

5.2.1 Initialization Phase

Initialization Algorithm. The algorithm runs on AA. It generates the global public parameters $PP = (n, X, q, m, s)$, Wherein, $n = n(\lambda)$ denotes the dimension of LWE, $X = X(\lambda) = D_{,\sqrt{n}}$ denotes an Error distribution,$q = q(n) = \tilde{O}(n^2 d_{\max})^{d_{\max}} n$ denotes the module of LWE, $m = m(n) = O(n \log q)$ denotes the number of samples, $s = s(n) = O(\sqrt{n \log q})$ denotes a Gaussian parameter. $B = B(n) = O(n)$ denotes an order of Error, $d_{\max}$ denotes the maximum number of times the algorithm $Recode(.)$ runs. That is, the depth of $Recode(.)$.

Setup Algorithm. The algorithm runs on AA. It has four steps.

Step 1: AA generates their public keys $\{APK_i\}$ for all attributes.

$$APK_i = \mathbf{pk}_i = \mathbf{A}_i,\ \mathbf{A}_i \in {}_q^{n\times m}.$$

Step 2: AA calculates the circuit output public key PK_{out} for different access control policies.

$$PK_{out} = pk_{out} = A_{out},\ A_{out} \in Z_q^{n\times m}.$$

Step 3: AA issues up-to-date a public and private key pair for each consumer.

$$\begin{aligned} GPK_{sub} &= A_{sub}, \mathbf{A}_{sub} \in {}_q^{n\times m}, \\ USK_{sub} &= T_i^{sub} i \in [0, sub\_attr] \end{aligned}$$

Wherein, $T_i^{sub} \in {}_q^{m\times m}$ is the trapdoor matrix of $\mathbf{A}_i + \mathbf{A}_{sub} \in {}_q^{n\times m}$, which is generated by calling the trapdoor generation algorithm TrapGen [3], *sub_attr* denotes the number of consumer attributes.

It is worth noting that we bind USK_{sub} with the latest identity private key. The purpose of this design is to force each consumer to use the newly issued attribute private key.

Step 4: AA releases $\{APK_i\}$ and PK_{out} to providers, transmits GPK_{sub} to semi-trusted edge routers, and issues USK_{sub} to the consumer using the secret exchange protocol.

5.2.2 Data Preparation Phase

EncryptData Algorithm. The algorithm runs on providers. It encrypts M to cm using the symmetric encryption algorithm.

BindingPolicy Algorithm. The algorithm runs on the provider. The encryption key $K_{Content}$ is encrypted with the CP-ABE method to obtain cp, $cm||cp$ is

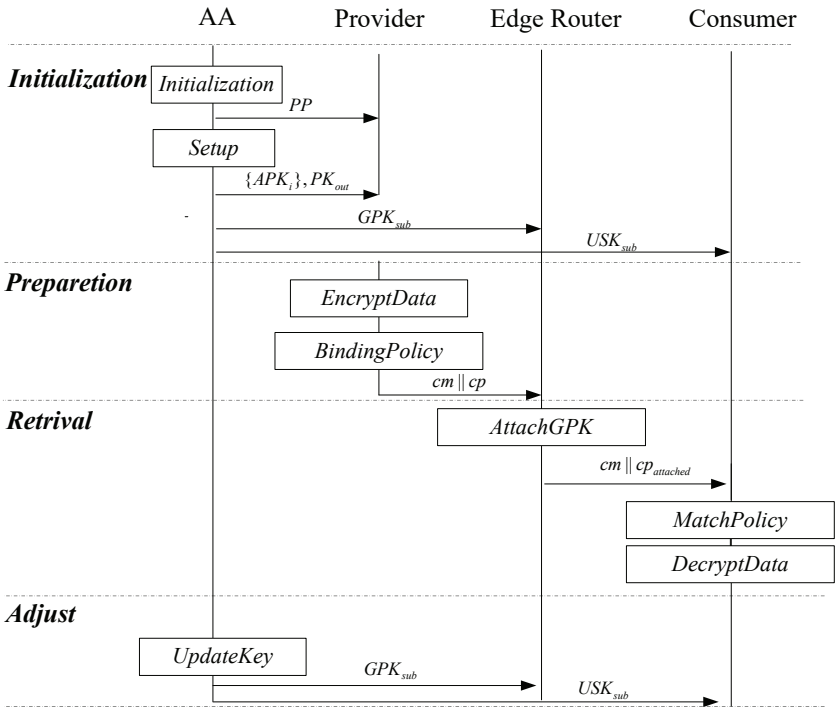


Figure 2: The framework of the proposed scheme

uploaded by the provider to the edge router. The CP-ABE encryption process is as follows.

$$\begin{aligned} cp &= (\varphi_1, \varphi_2, \cdots, \varphi_N, \Gamma, \mathbf{s}), \\ \varphi_i &= Encode(\mathbf{pk}_i, \omega), \\ \Gamma &= E(Encode(\mathbf{pk}_{\text{out}}, \omega), K_{Content}) \end{aligned}$$

Wherein,$\mathbf{s} \in {}_q^n$ is a random vector picked in $Encode(\mathbf{pk}, \omega)$. The provider uploads $cm||cp$ onto edge routers.

5.2.3 Data Retrieval Phase

AttachGPK Algorithm. The algorithm runs on edge routers.

Before the edge router transmits content to the consumer, it uses the consumer's latest GPK_{sub} to update cp, making $cp_{attached}$ different for each consumer. The update process is as follows: $cp_{attached} = (\varphi_1^{'}, \varphi_2, \cdots, \varphi_{sub\_attr}^{'}, \Gamma)$,$\varphi_i^{'} \leftarrow \varphi_i + (GPK_{sub})^{\mathrm{T}} \cdot \mathbf{s}$ The edge router then sends $cm||cp_{attached}$ to the consumer.

MatchPolicy Algorithm. The algorithm runs on a consumer.

$$\begin{aligned} rk &= RekeyGen(\{\varphi_i\}, USK_{sub}, \mathbf{pk}_{\text{out}}), \\ \varphi_{\text{out}} &= Recode(rk, \varphi_1, \varphi_2, \cdots, \varphi_{sub\_attr}) \end{aligned}$$

Then the consumer calculates $D(\varphi_{\text{out}}, \Gamma)$ to get $K_{Content}$.

DecryptData Algorithm. The algorithm runs on a consumer. It runs a symmetric decryption algorithm to change cm into M using $K_{Content}$.

5.2.4 Permission Adjustment Phase

UpdateKey Algorithm. Only one algorithm is run at this phase: the $UpdateKey$ algorithm, which runs on AA to generate the latest identity public key GPK_{sub} and the corresponding private key USK_{sub}:

$$\begin{aligned} GPK_{sub} &= A_{sub}, \mathbf{A}_{sub} \in {}_q^{n \times m}, \\ USK_{sub} &= T_i^{sub}, i \in [0, sub\_attr]; \end{aligned}$$

Wherein, $T_i^{sub} \in {}_q^{m \times m}$ is the trapdoor matrix of $\mathbf{A}_i + \mathbf{A}_{sub} \in {}_q^{n \times m}$, and sub_attr denotes the number of a consumer attributes. Finally, AA issues GPK_{sub} and USK_{sub} to the edge router and the consumer separately.

6 Security Analysis

Analysis of Data Confidentity. When the consumer requests the content, the edge router will send the $cm||cp_{attached}$ associated with GPK_{sub} to the consumer. If the attribute set carried by the consumer matches $cp_{attached}$, the consumer can decrypt $cp_{attached}$ and obtain the symmetric encryption key

$K_{content}$, so that the content can be decrypted from cm by running the symmetric decryption algorithm. Non-authorized consumers or edge routers cannot read the content even if they get cm and $cp_{attached}$.

Analysis of Forward and Backward Security.

Forward Security Analysis. New consumers can access contents when their attribute set is satisfied with$cp_{attached}$.

Backward Security Analysis. when a consumer with revoked attributes accesses contents, the edge router updates $cp_{attached}$, which is associated with the latest identity key GPK_{sub}, so that the consumer cannot continue to use the old private key USK_{sub}. The detailed analysis starts from updating cp:

$$\begin{aligned}\varphi_i + (GPK_{sub})^{\mathrm{T}} \cdot s &= \mathbf{A}_i{}^{\mathrm{T}}\mathbf{s} + \mathbf{e} + \mathbf{A}_{sub}{}^{\mathrm{T}}\mathbf{s} \\ &= (\mathbf{A}_i{}^{\mathrm{T}} + \mathbf{A}_{sub}{}^{\mathrm{T}}) \cdot \mathbf{s} + \mathbf{e} \\ &= (\mathbf{A}_i + \mathbf{A}_{sub})^{\mathrm{T}} \cdot \mathbf{s} + \mathbf{e}\end{aligned}$$

In addition, AA runs the $UpdateKey$ algorithm to generate the latest private key $USK_{sub} = T_i^{sub}\}$ after the permission adjustment for consumers, where USK_{sub} is exactly the trapdoor matrix of $\mathbf{A}_i + \mathbf{A}_{sub} \in {}_q^{n\times m}$. Therefore, consumers whose attribute set meets cp can still access the ciphertext after the permission adjustment, that is, the scheme is still correct.

Analysis of Collusion Resistance. In the $AttachGPK$ algorithm, the edge router binds φ_i with $(GPK_{sub})^{\mathrm{T}} \cdot \mathbf{s}$ every time when it accepts the consumer's access request, and the consumer cannot get φ_i from $\varphi_i^{'}$. On the other hand, $cp_{attached}$ get by each consumer is associated with its own GPK_{sub}, so that even if two users merge their USK_{sub}, they won't get more privileges.

7 Efficiency Analysis and Simulation

7.1 Storage Cost

Table 1 shows the storage overhead on each entity of the proposed scheme. Wherein, n_{attr} denotes the order of attribute set, n_{sub} denotes the order of consumer set, N denotes the number of matrices recorded in each transformation table, i.e. the number of inputs in each circuit. It is also the number of attributes in the policy bound by the provider. n_{sub_attr} denotes the number of attributes in a consumer private key. $|A_{i,j}|$ denotes the storage cost per matrix.

In our scheme, the storage cost on AA derives from $\{APK_i\}$. The storage cost on the cloud derives mainly from GPK_{sub} and the transformation table $cp = (\varphi_1, \varphi_2, \cdots, \varphi_N, \Gamma)$ associated with the ciphertext. The storage cost on a provider derives from both $\{APK_i\}_{pub}$ and PK_{out} which are needed to run the BindingPolicy algorithm. The consumer's storage cost derives from the private key USK_{sub}.

Table 1: Storage costs on each entities

AA	Edge router	Provider	Consumer
n_{attr}	$n_{sub} + N + 1$	$N + 1$	n_{sub_attr}

7.2 Computation Cost

Table 2 shows the computational cost on each entity of the proposed scheme. Wherein, $|SampleD|$ denotes the computational cost from the Gaussian sampling function. $trans$ denotes the computational cost from the matrix transpose. $multi$ denotes the computational cost from the matrix multiplication. We ignore the cost from matrix addition.

Table 2: Computational costs of core algorithms

Algorithm	Computation Cost
Setup	$(n_{attr} + n_{sub}n_{attr})\|SampleD\|$
BindingPolicy	$(N+1)(trans+multi)$
AttachGPK	$trans+multi$
Matchpolicy	$(N+1)\|SampleD\| + (N+1)multi$ $+(N+1)trans$
UpdateKey	$(n_{sub\_attr}+1)\|SampleD\|$

In the proposed scheme, the computational cost of the Setup algorithm is mainly the sampling of the matrix $\{\mathbf{A}_i\}$. It is necessary to run the $SampleD$ algorithm for $n_{attr}+n_{sub}$ times, because the generation of the public key and private key for each attribute needs n_{attr} times and the generation of the identity public key GPK_{sub} and attribute private key USK_{sub} for each consumer needs n_{sub} times. The computational cost of the $BindingPolicy$ algorithm is caused mainly by the circuit coding . It is necessary to record the operational process when the $Encode$ algorithm runs for $N + 1$ times and the encryption algorithm E once for each translation table. The calculation cost of the $MatchPolicy$ algorithm is mainly from matching $cp_{attached}$, and it needs to run the $ReKeyGen$ algorithm once and the $Recode$ algorithm once and the decryption Algorithm D once. The computational cost of the $UpdateKey$ algorithm is to re-issue the consumer's identity public key GPK_{sub} and attribute private key USK_{sub}.

7.3 Simulation

Table 3 shows the simluation results. The simulation test is done on OpenSUSE 64bit running on the VMware Workstation 15 Pro with 6 processors and 7.5G RAM. We use FLINT (fast library for number theory), MPFR (multiple precision floating point reliable library) and MPIR(multiple precision integer and rational library) in our simulation.

Table 3: Simluation of core algorithms (ms)

Algorithm	Time
Encoding	33.83
E	0.43
D	1.99
SampleD	1609.50

The n (dimesion of LWE) is set to be 512. The m (number of samples) is set to be $poly(n)$. The algorithm *SampleD* uses the Nearest-Plane method.

8 Conclusion

Based on the LWE assumption, we put forward an NDN data sharing scheme supporting direct consumer permission revoking in allusion to requirements for NDN data sharing of loose coupling, fine-grained control, consumer rights, dynamic management. Our scheme has the following advantages: 1) the authorized consumer can only access the authorized part of contents, realizing the fine-grained access control; 2) ABE scheme is realized using matrix calculation by instantiating an LWE-based MOR primitives; 3) Consumer permission can be adjusted flexibly through binding ciphertext with the latest GPK_{sub} on the edge router.

Acknowledgments

This work was supported by National Science Foundation of China (No. 62062045).

References

[1] S. Agrawal, M. Maitra, and S. Yamada, "Attribute based encryption (and more) for nondeterministic finite automata from lwe," in *Annual International Cryptology Conference*. Springer, 2019, Conference Proceedings, pp. 765–797.

[2] P. Datta, I. Komargodski, and B. Waters, "Decentralized multi-authority abe for dnfs from lwe," in *Annual International Conference on the Theory and Applications of Cryptographic Techniques*. Springer, 2021, Conference Proceedings, pp. 177–209.

[3] S. Gorbunov, V. Vaikuntanathan, and HoeteckWee, "Attribute-based encryption for circuits," in *45fth annual ACM symposium on Theory of computing (STOC)*. New York: ACM, 2013, Conference Proceedings, pp. 545–554.

[4] S. Jiaoli, H. Chuanhe, H. Kai, S. Xieyang, and H. Chao, "An access control method supporting multi-user collaborative edit in cloud storage," *Journal of Computer Research and Development*, vol. 54, no. 7, pp. 1603–1616, 2017.

[5] X. Jie, W. Qiaoyan, L. Wenmin, and J. Zhengping, "Circuit ciphertext-policy attribute-based hybrid encryption with verifiable delegation in cloud computing," *IEEE Transactions on Parallel and Distributed Systems*, vol. 27, no. 99, pp. 119–129, 2015.

[6] L. Jiguo, Y. Wei, H. Jinguang, Z. Yichen, and S. Jian, "User collusion avoidance cp-abe with efficient attribute revocation for cloud storage," *IEEE Systems Journal*, vol. 12, no. 2, pp. 1767–1777, 2018. [Online]. Available: http://ieeexplore.ieee.org/ielx7/4267003/4357939/07867082.pdf?tp=&arnumber=7867082&isnumber=4357939

[7] Z. Kai, M. Jianfeng, L. Hui, Z. Junwei, and Z. Tao, "Multi-authority attribute-based encryption with efficient revocation," *Journal on Communications (In chinese)*, vol. 38, no. 3, pp. 83–91, 2017.

[8] S. Kim, "Multi-authority attribute-based encryption from lwe in the ot model," *Cryptology ePrint Archive*, 2019.

[9] H. Kun, C. Jing, Z. Yu, D. Ruiying, X. Yang, H. M. Mehedi, and A. Abdulhameed, "Secure independent-update concise-expression access control for video on demand in cloud," *Information Sciences*, vol. 387, no. 2017, pp. 75–89, 2017. [Online]. Available: http://dx.doi.org/10.1016/j.ins.2016.08.018http://ac.els-cdn.com/S0020025516305904/1-s2.0-S0020025516305904-main.pdf?_tid=9d8c37d6-3aa9-11e7-94bd-00000aab0f26&acdnat=1494988762_ebc2c9a0683b6b3e27ad196691ebd8e9

[10] L. Ming, Y. Shucheng, Z. Yao, R. Kui, and L. Wenjing, "Scalable and secure sharing of personal health records in cloud computing using attribute-based encryption," *IEEE Transactions on Parallel and Distributed Systems*, vol. 24, no. 1, pp. 131–143, 2013. [Online]. Available: ⟨GotoISI⟩://000312837400012

[11] Named Data Network, "NDN protocol design principles," Aug. 24, 2022. (https://named-data.net/project/ndn-design-principles/)

[12] L. Qin, W. Guojun, and W. Jie, "Time-based proxy re-encryption scheme for secure data sharing in a cloud environment," *Information Sciences*, vol. 258, no. 2014, pp. 355–370, 2014.

[13] O. Regev, "On lattices, learning with errors, random linear codes, and cryptography," *Journal of the ACM*, vol. 56, no. 6, pp. 1–40, 2009. [Online]. Available: http://dx.doi.org/10.1145/1568318.1568324http://dl.acm.org/citation.cfm?doid=1568318.1568324

[14] R. Tsabary, "Fully secure attribute-based encryption for t-cnf from lwe," in *Annual International Cryptology Conference*. Springer, 2019, Conference Proceedings, pp. 62–85.

[15] G. Wang, Z. Liu, and D. Gu, "Ciphertext policy attribute-based encryption for circuits from lwe assumption," in *International Conference on Information and Communications Security*. Springer, 2019, Conference Proceedings, pp. 378–396.

[16] L. Ximeng, M. Jianfeng, X. Jinbo, L. Qi, Z. Tao, and Z. Hui, "Threshold attribute-based encryption with attribute hierarchy for lattices in the standard model," *IET Information Security*, vol. 8, no. 4, pp. 217–223, 2014.

[17] Y. Xixi and M. Hui, "Ciphertext policy attribute-based encryption scheme supporting direct revocation," *Journal on Communications (In chinese)*, vol. 37, no. 05, pp. 44–50, 2016.

Biography

Jiaoli Shi received the Ph.D. degree from the School of Computing, Wuhan University, in 2017. Since 2012, she has been an Assistant Professor at the school of computer and big data science, Jiujiang University. Her research interests include Cloud security, ICN security, SDN and CDN. Dr. Jiaoli twice won the third prize of Doctoral Forum of School of Computer Science, Wuhan University for Excellence in 2014 and 2015.

Anyuan Deng received the B.Sc. degree in computer science from Jiangxi Normal University, Nanchang, China, in 1995, M.Sc. degree in computer science from Huazhong University of Science and Technology, Wuhan, China, in 2006. Since 2003, He has been a professor at the school of computer and big data science, Jiujiang University, Jiujiang, China. His research interests include Cloud security, ICN security, Software Engineering and CDN.

Chao Luo is an undergraduate student in the school of computer and big data science, Jiujiang University. His research interests include Cloud security, Privacy protection.

Shimao Yao received the Ph.D. degree from the Kunsan National University, Kunsan, South Korea, 2021. He has been a Lecturer at the school of computer and big data science, Jiujiang University, Jiujiang, China. His research interests include Cloud security, Applied cryptography.

Kai He received the B.S. degree in computer science from Wuhan Textile University, Wuhan, China, in 2010, and the Ph.D. degree from Wuhan University, Wuhan, China, in 2016. Since 2016, He has been a Lecturer at the School of Mathematics and Computer, Wuhan Textile University, Wuhan, Hubei, China. From Jan. to June 2015, he was a visiting Student at the University of Calgary. His research interest includes the Cloud security, auction, *et al.*.

Study on the Evidence Collection for Network Security Intrusion Detection

Xindong Wang
(Corresponding author: Xindong Wang)
Shaanxi Police College, Xi'an, Shaanxi, China
No. 199, Qiyuan Second Road, Weiyang District, Xi'an, Shaanxi 710021, China
Email: muxin090786@163.com
(Received May 23, 2019; Revised and Accepted Aug. 13, 2022; First Online Oct. 15, 2022)

Abstract

The expansion of the Internet not only brings convenience but also increases the risk of network information security. This paper briefly introduced the software-defined network (SDN) technology and the SDN-based network intrusion detection and evidence collection system. The system had a dynamic priority scheduling strategy for intrusion detection and evidence collection response tasks. The designed network intrusion detection and evidence collection system were simulated in a small SDN built in a laboratory and compared with the other system adopting a static priority task scheduling strategy. The results suggested that the designed network intrusion detection and evidence collection system effectively collected suspicious traffic; compared with the network intrusion detection and evidence collection system using the static priority task scheduling strategy, the designed detection and evidence collection system was comparable in terms of detection accuracy but had more advantages in terms of evidence collection time and switch transmission rate.

Keywords: Evidence Collection; Intrusion Detection; Network Security; Software-Defined Network

1 Introduction

With the globalization of information and the rapid development of the Internet, companies, governments, and individuals are cooperating on the Internet to enhance work efficiency and convenience and enrich people's leisure time [1]. Networks have two advantages, open and sharing. On the one hand, the two advantages give the network various resources, facilitating people's life; on the other hand, cyber criminals can also take advantage to launch an invasion against enterprises, governments, or individuals through the network [2].

In recent years, the popularity of mobile terminals that can access the Internet has also diversified the ways of cybercrime, which poses greater security risks to users than the convenience it brings. Therefore, in order to protect users' network security, it is necessary to make effective protection against intrusion data and record them in the process of protection to facilitate the traceability and forensics of the intrusion data [3]. Lu *et al.* [1] proposed a new model based on the optimized back-propagation neural network (BPNN) and Dempster-Shafer theory to detect intrusion data and verified the effectiveness of the method through experimental simulations.

Jing *et al.* [5] proposed a network intrusion detection method based on associative deep learning and found from simulation results that the method had a high average detection rate and average error detection rate for unknown intrusions and attacks. Xie *et al.* [6] proposed an online distributed intrusion detection model based on a cellular neural network and found from the experiment on the KDD CUP 99 dataset that the method was feasible and effective. This paper briefly introduced the software-defined network (SDN) technology and the SDN-based network intrusion detection and evidence collection system, simulated the designed system in a small SDN built in a laboratory, and compared it with an intrusion detection and evidence collection system adopting another task scheduling strategy.

2 Detection and Evidence Collection of Network Intrusion Data

2.1 Introduction of SDN Technology

The expansion of the Internet has greatly facilitated people's lives, but it has also provided a corresponding platform for law-breakers. The Internet's development brings not only convenience but also network security problems. For example, law-breakers will use the vulnerability of Internet protocols to hijack normal users' computers as "broiler chickens" and launch distributed denial of service (DDoS) attacks consisting of a large amount of malicious traffic to target users [7]. In the face of increasing network security problems, the artificial extension of con-

figurations for network security devices is difficult due to the complex structural system of the traditional network, making the efficient automatic response difficult [8]. SDN technology has emerged as a new network security protection strategy.

Compared with the traditional network structure, SDN has only three structures: application layer, control layer, and data layer, which is relatively simpler in network maintenance. In SDN, data transmission and control are separated, i.e., data are transmitted using SDN switches in the data layer, but the switches are only responsible for the transmission of data and do not have the function of deciding the destination of data transmission. The function of controlling data transmission is realized in the control layer. The control layer receives tasks from the application layer and sends corresponding control commands to the switch in the data layer, and the switch opens and closes ports according to the commands [9].

Since SDN separates control and transmission, it is much easier to extend the network. The switch in the data layer is only responsible for data transmission, so it is not necessary to consider the structural changes of the data layer when extending the SDN, but only to add the corresponding extended interfaces in the control layer and the corresponding interactive interface software in the application layer.

2.2 SDN-based Network Intrusion Detection and Evidence Collection

As introduced in the previous section on SDN, the network structure has good scalability after separating the control and transmission of data, but even if the network structure is optimized, malicious attacks on the Internet will not be reduced. Due to the characteristics of the SDN structure, DDoS is a common form of attack [10]. During a DDoS attack, a large amount of false or meaningless traffic data is used to make the SDN control generate a large number of requests and flow entries, thus taking up the computing resources of the controller and switches and eventually bringing the network down.

In order to resist DDoS attacks, SDN needs to conduct the corresponding detection in the process of data transmission to distinguish normal data from abnormal data and then intercept the abnormal data. At the same time, the data judged to be abnormal are collected and stored, which can be used as evidence for recourse or a template to further improve the identification performance of the abnormal data detection algorithm [11].

Figure 1 shows the basic structure of the SDN-based network intrusion detection and evidence collection system. Since the network intrusion detection and evidence collection system is based on SDN, its basic structure is also divided into three layers. The application layer contains a DDoS interception function and an evidence collection function. These two functions in the application layer have a user-oriented interface so that users can intuitively see the DDoS interception results and alerts and

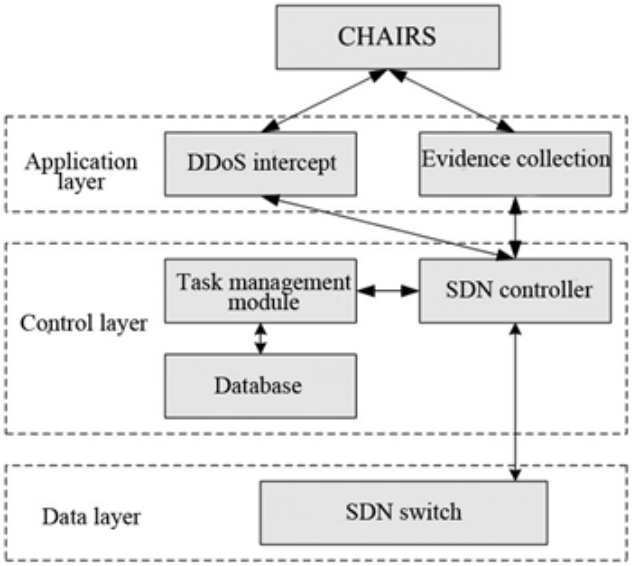


Figure 1: Basic structure of the SDN-based network intrusion detection and evidence collection system

the evidence collection report of abnormal data [12]. The full name of "CHAIRS" in Figure 1 is cooperative hybrid aided incidence response system, which is a distributed network emergency management system that mainly receives alerts from the active security defense system in SDN and responds to them. It is a distributed network emergency management system that receives and responds to alerts from the active security defense system in the SDN.

The control layer includes the SDN controller, task management module, and database. The SDN controller is responsible for receiving requests from the application layer, converting them into flow entries, and sending them to the switch at the data layer. It also plays the role of switch monitoring in the process of network intrusion detection and evidence collection [13]. The database not only caches the task requests but also stores the suspicious messages obtained from the evidence collection for storage. The task management module is responsible for managing the tasks issued by the application layer and allocating computing resources according to the task priority.

The data layer is composed of SDN switches, whose role is to transmit data and follow the flow entries issued by the SDN controller when transmitting data [14]. Data transmission in SDN is entirely the responsibility of the switch. The control layer can control the direction of data transmission by simply controlling the opening and closing of switch ports. In other words, the data traffic transmitted in the SDN structured network does not directly pass through the control and application layers, which makes the security of the control and application layer devices improved, but DDoS does not directly damage the control and application layer devices but uses junk

traffic to occupy computing resources, which is considered an attack method for the characteristics of SDN structure. Therefore, SDN networks still need to make active protection against DDoS intrusion attacks and collect evidence.

2.3 The Process of Network Intrusion Detection and Evidence Collection

The network intrusion detection and evidence collection process based on the SDN structure is shown in Figure 2. In this process, the SDN controller needs to perform tasks including opening and closing switch ports and collecting evidence of suspicious traffic transmitted by the switch. Every task is quite heavy, but the computational resources of the whole system are limited, so it is impossible to process all tasks together. Therefore, it is necessary to follow the scheduling policy to enable the tasks to be executed. The traditional scheduling strategy assigns different priority fields to different tasks and executes the tasks in the order of priority, but this fixed priority scheduling strategy will lead to serious polarization of computing resources between low priority and high priority tasks, so this paper adopts a dynamic priority scheduling strategy to allocate queues [15], and the specific steps are shown in Figure 2.

1) The server in the SDN controller responsible for monitoring the switch traffic collects the feature fields of the traffic data in the switch.

2) The collected traffic feature field is uploaded to the DDoS interception module in the application layer. The DDoS attack is detected using the corresponding DDoS detection algorithm. It returns to Step 1 if no DDoS attack is detected; if a DDoS attack is detected, an early warning is sent to CHAIRS through the system interface.

3) The CHAIRS receives the alert and issues response tasks to the DDoS interception module and evidence collection module in the application layer through the system interface according to the set script.

4) After receiving the response tasks, the DDoS interception and evidence collection modules in the application layer both generate the corresponding control commands and send the task commands to the SDN controller through the application programming interface (API).

5) After receiving the task commands, the SDN controller judges whether the current computing resources of the system can support the task command according to the scheduling policy of the task management module. If it can, the task command is put into the execution queue; if not, it is put into the waiting queue.

6) The task command is selected from the waiting queue in priority order. Whether the remaining system resources can support the execution of the task command is determined. If it can, the task command is put into the execution queue. If not, whether the task can be replaced by a lower priority task in the execution queue to obtain computer resources is determined. If it can, the two tasks are exchanged; if not, the task is put back into the waiting queue after adding one to its priority. It is recorded as one scheduling cycle when Step 6 is cycled once.

7) In every scheduling cycle, the SDN controller receives tasks from the execution queue in priority order, generates the corresponding flow entries, and sends them to the SDN switch through the OpenFlow protocol.

8) The SDN switch controls the data transmission based on the received flow entries, including the closing or opening of the attacked ports and the transmission of messages of suspicious traffic to the corresponding database. Finally, it returns to Step 1.

3 Simulation Experiments

3.1 Experimental Setup

The simulation experiment was conducted in a small SDN built in a laboratory. Figure 3 shows the basic architecture of the SDN for the simulation experiment. The whole network intrusion detection and evidence collection system had four servers and one switch. One server was used as the SDN controller, which took the role of the application and control layers, and one server was used as the database for storing messages of suspicious traffic collected. The remaining two servers served as regular servers for simulating two users who transmitted data. The switch played the role of data transmission. The switch was connected to all the servers, but the flow entry information was exchanged between the SDN controller and the switch through the OpenFlow protocol. The switch was used as a transit for data interaction between the regular servers, and it was a one-way connection from the switch to the database.

In the whole process of network intrusion detection and evidence collection, the transmission traffic of the switch was monitored using the switch monitoring module in the SDN controller, and the traffic data was detected using the intrusion detection algorithm. A BPNN was used to warn the traffic data. When the intrusion data were detected, a warning was issued. After receiving the warning, the CHAIRS created response tasks, and then the task management module in the SDN controller scheduled and assigned the tasks, converted them to flow entries in order, and sent them to the switch. The switch transmitted the data according to the flow entries, including intercepting the malicious data and collecting them as evidence.

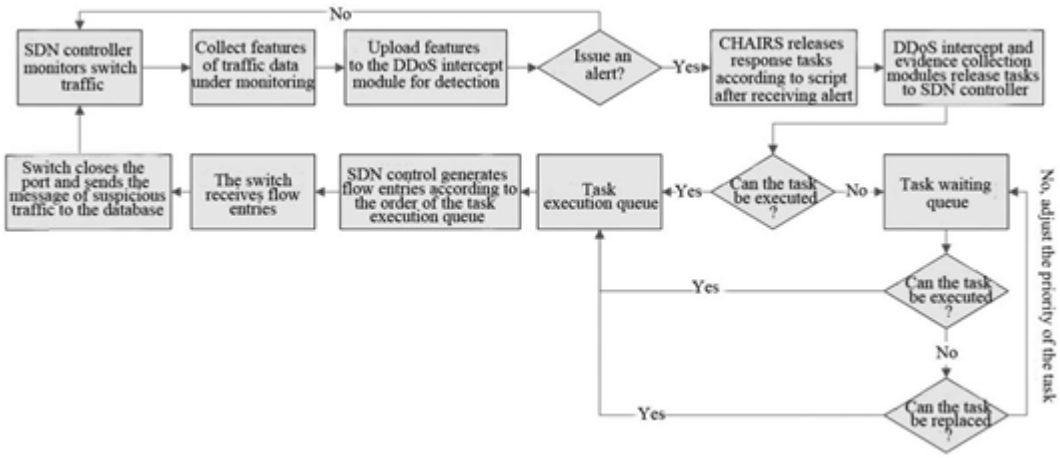


Figure 2: The basic process of network intrusion detection and evidence collection

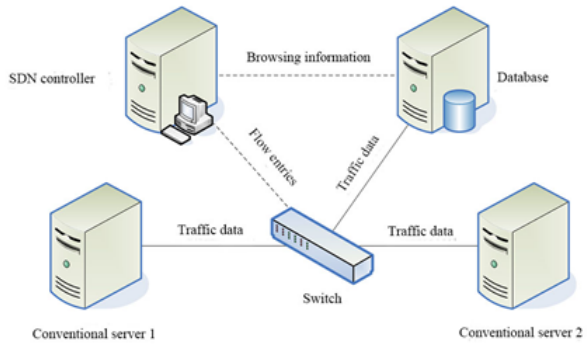


Figure 3: Architecture diagram of the SDN-based network intrusion detection and evidence collection system

3.2 Experimental Projects

Two thousand packets of data of different sizes were sent from conventional server 1 to conventional server 2, and 500 were treated as anomalous packets, i.e., intrusion data. The designed network intrusion detection and evidence collection system was used to detect packets and collect evidence. In order to further verify the effectiveness of the improved task scheduling strategy, a network intrusion detection forensics system adopting the conventional static priority task scheduling strategy was also simulated for comparison. The system used for comparison was architecturally consistent with Figure 3, but the only difference was the task scheduling strategy used by the task management module within the SDN controller.

3.3 Experimental Results

The designed network intrusion detection and evidence collection method was used in the simulation experiment to detect and collect evidence from the data transmitted between conventional servers 1 and 2. As the amount of data acting as abnormal data was large during the experiment, the number of response tasks generated during the experiment was also very large. Limited by space, only the feedback result of evidence collection response task number 3 is shown here, as shown in Figure 4. It was seen from the feedback result that the evidence collection task number of this feedback result was 3, which started at 18:42:41 on June 13, 2022, and ended at 18:55:36 on June 13, 2022, and 1.26 MB of suspicious messages were successfully collected, which triggered 365 alerts in the transmission process.

To verify the performance of the proposed network intrusion detection and evidence collection method, it was compared with a network intrusion detection and evidence collection system with the same structure but a different task scheduling strategy. Table 1 shows the performance test results of network intrusion detection and evidence collection under two task scheduling strategies. The P value in the comparison of the detection accuracy between the traditional static priority task scheduling strategy and the dynamic priority task scheduling strategy was 0.165, i.e., the difference was not significant. The P value of the comparison of the average time of evidence collection was 0.01, and the method adopting the dynamic priority task scheduling strategy was faster in collecting suspicious data. The P value in the comparison of the average data transmission rate in the switch was 0.01, and the detection and evidence collection method that adopted the dynamic priority task scheduling strategy had higher switch data transmission rate.

The reason for these results was analyzed. The two network intrusion detection and evidence collection methods only differed in their task scheduling strategies. For the network intrusion detection and evidence collection method, the accuracy of network intrusion data detection depended on the intrusion detection algorithm. Both methods used a BPNN to detect traffic data in the simulation experiment, so they were comparable in terms of detection accuracy. In terms of the evidence collec-

Table 1: Network intrusion detection forensics performance under two task scheduling policies

	Detection accuracy/%	The average time spent on evidence collection/s	The average data transmission rate of the switch MB/s
Traditional static priority task scheduling strategy	98.6	265	523
Dynamic priority task scheduling strategy	98.5	203	869
P value	0.165	0.01	0.01

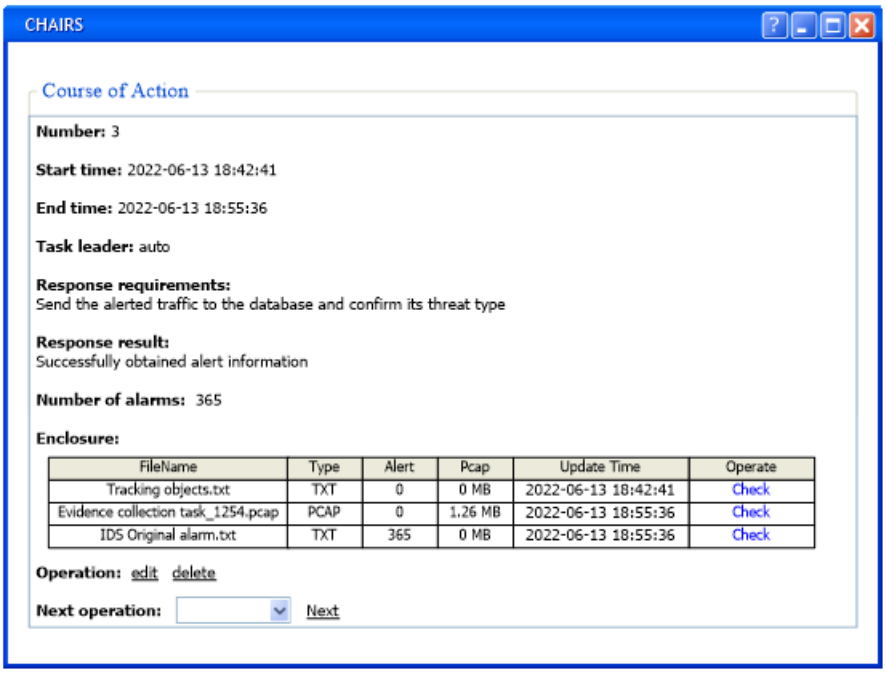


Figure 4: The feedback result of forensic task number 3

tion time and switch data transmission rate, the method that adopted the dynamic priority task scheduling strategy was more advantageous. The reason is as follows. The traditional static priority task scheduling strategy processed tasks in priority order, but if the new response task had a high priority for a long time, tasks with low priorities in the waiting queue would not be treated for a long time, i.e., new tasks with high priorities would cut in line, affecting the actual task processing. The dynamic priority task scheduling policy processed tasks in priority order as a whole, but low-priority tasks tried to replace with lower-priority tasks in the execution queue, and if that did not work, they would adjust their priorities upward to be selected in the execution queue in the next scheduling cycle.

4 Conclusion

This paper briefly introduced the SDN technology and the SDN-based network intrusion detection and evidence collection system, which had a dynamic priority scheduling strategy for intrusion detection and evidence collection response tasks. The proposed network intrusion detection and evidence collection system was simulated in a small SDN built in a laboratory, and it was compared with the network intrusion detection and evidence collection system adopting a static priority task scheduling strategy. The following findings were obtained. The network intrusion detection and evidence collection system could effectively collect suspicious traffic in the network. The intrusion detection and evidence collection systems adopting different task scheduling strategies were not significantly different in the detection accuracy of suspicious data, but the intrusion detection and evidence collection system adopting the dynamic priority scheduling strategy was more advantageous in the evidence collection time and switch transmission rate.

References

[1] S. Khan, A. Gani, A. Wahab, M. Shiraz, I. Ahmad, "Network forensics: Review, taxonomy, and open challenges," *Journal of Network & Computer Applications*, vol. 66, no. May, pp. 214-235, 2016.

[2] C. Jain, A. K. Saxena, "General study of mobile agent based intrusion detection system (IDS)," *Journal of Computer and Communications*, vol. 4, no. 4, pp. 93-98, 2016.

[3] G. Xu, "Research on network intrusion detection method based on machine learning," *Journal of Physics: Conference Series*, vol. 1861, no. 1, pp. 1-6, 2021.

[4] C. Lu, L. Yue, M. Ma, N. Li, "A hybrid NIDS model using artificial neural network and D-S evidence," *International Journal of Digital Crime & Forensics*, vol. 8, no. 1, pp. 37-50, 2016.

[5] L. Jing, W. Bin, "Network intrusion detection method based on relevance deep learning," in *International Conference on Intelligent Transportation*, pp. 237-240, 2016.

[6] K. Xie, Y. Yang, Y. Xin, G. Xia, "Cellular neural network-based methods for distributed network intrusion detection," *Mathematical Problems in Engineering*, vol. 2015, no. pt.3, pp. 1-10, 2015.

[7] G. Fanani, I. Riadi, "Analysis of digital evidence on denial of service (DoS) attack log based," *Buletin Ilmiah Sarjana Teknik Elektro*, vol. 2, no. 2, pp. 70, 2020.

[8] S. Kalime, "Efficient network intrusion detection system using Boyer Moore algorithm index termsnetwork intrusion detection system; packet capturing module, Boyer-Moore," *International Journal of Research*, vol. 4, no. 17, pp. 1083, 2017.

[9] F. Shang, D. Zhou, C. Li, H. Ye, Y. Zhao, "Research on the intrusion detection model based on improved cumulative summation and evidence theory for wireless sensor network," *Photonic Network Communications*, vol. 37, no. 2, pp. 1-12, 2018.

[10] Z. Wang, "Network intrusion detection by using combination optimizing features and classifier parameters," *Journal of Nanjing University of Science & Technology*, vol. 41, no. 1, pp. 59-64, 2017.

[11] D. Gugelmann, F. Gasser, B. Ager, V. Lenders, "Hviz: HTTP(S) traffic aggregation and visualization for network forensics," *Digital Investigation*, vol. 12, pp. S1-S11, 2015.

[12] L. Duan, F. Yu, L. Zhan, "An improved fuzzy C-means clustering algorithm," in *International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery*, pp. 44-46, 2016.

[13] P. Nayak, A. Devulapalli, "A fuzzy logic-based clustering algorithm for WSN to extend the network lifetime," *EEE Sensors Journal*, vol. 16, no. 1, pp. 137-144, 2015.

[14] T. Yoshioka, S. Karita, T. Nakatani, "Far-field speech recognition using CNN-DNN-HMM with convolution in time," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 4360-4364, 2015.

[15] N. Khamphakdee, N. Benjamas, S. Saiyod, "Improving intrusion detection system based on Snort rules for network probe attacks detection with association rules technique of data mining," *Journal of ICT Research & Applications*, vol. 8, no. 3, pp. 234-250, 2015.

Biography

Wang Xindong was born in Hanzhong City, Shaanxi Province. He obtained a master's degree from Chang'an University in June 2011. He is an engineer and is working at Shaanxi Police College. He is interested in the network security of industrial control systems.

A Modified ZigZag Transform Method and Its Application in Image Encryption

Chunming Xu and Yong Zhang
(Corresponding author: Chunming Xu)

School of Mathematics and Statistical, Yancheng Teachers University, P. R. China
No.50, Kaifang Avenue, Yancheng 224002, P. R. China
Email:ycxcm@126.com

(Received Mar. 2, 2022; Revised and Accepted Oct. 8, 2022; First Online Oct. 15, 2022)

Abstract

This paper proposes a novel image encryption algorithm based on ZigZag transform and chaotic system with the scrambling-diffusion structure. Firstly, we propose a modified ZigZag scrambling transform method to overcome the shortcomings of the traditional ZigZag transform. Secondly, the modified ZigZag scrambling method is adopted to scramble the image coefficient matrix successfully. Finally, the diffusion method is used to encrypt images to obtain the final cipher image. In addition, the SHA256 hash function value of the original image is generated to calculate the parameters for the initial values of the chaotic system, which enhances the correlation between the algorithm and the plain image and makes the proposed encryption scheme resist the plaintext attacks. Experiments are tested on three classical images. The results of simulation experiments are evaluated using the histogram, correlation analysis, entropy, number of pixel change rate (NPCR), and unified average change intensity (UACI). The experimental results show that the proposed method is practical and feasible.

Keywords: Chaotic System; Image Encryption; Image Scrambling; ZigZag Transform

1 Introduction

With the development of computer network technology and the advent of the era of big data, a large number of information emerge. As an important carrier of information, image data plays an important role in information transmission. Many images contain valuable information and need to be kept secret during transmission. For example, in the political, economic, financial and other aspects, a lot of images need to be kept confidential to protect their values. However, due to the openness and vulnerability of the network, the security of the image is greatly threatened. The image can be encrypted before transmission to protect its useful information, so that the image encryption algorithm has received people's attention and become a research hotspot [1,7,9,10,13,16,20]. Different from text data, image data has not only a large amount of information but also a strong correlation between adjacent pixels. As a result, traditional data encryption algorithms such as AES, DES and RSA are not suitable for image data encryption. In 1989, Matthew introduced an effective image encryption algorithm based on chaotic sequence [11]. The proposed algorithm is based on scrambling-diffusion structure, which uses chaos to generate keystream, then scramble and encrypt the image to get an encrypted image. Since then, image encryption algorithm based on chaos has become the mainstream algorithm of image encryption due to that the chaotic sequence is easy to generate, and has the characteristics of strong sensitivity to initial conditions, pseudorandomness and complete reproducibility, which are suitable for image encryption.

For image scrambling and encryption, researchers have proposed many different algorithms, such as Arnold transform [3, 8, 14], DNA coding [4, 5, 15], Josephus transform [12, 22], etc. In [6], the author adopted ZigZag transform to encrypted the image. ZigZag transform is a simple and effective method to permutation image pixels. Its basic idea is to scan the elements of a matrix from the upper left corner to the lower right corner in ZigZag order to scramble the data. ZigZag transform has been widely used in image and video encryption because of its simple and low time complexity. However, an important problem with ZigZag transform is that it has periodicity. After certain rounds of transformation, the scrambled image will be recovered to the original image. In addition, after the ZigZag transform, the first and last elements will not be changed, which makes the scrambled image easy to be attacked, and the effect of scrambling is not very good.

Based on the above discussions, the image encryption method based on ZigZag transform is studied in this paper, and an improved ZigZag transform method is proposed and applied in image encryption. The improved ZigZag transform method can enhance the complexity of scrambling and the security of encryption algorithm.

Moreover, we use plaintext information to generate the initial value of the chaotic system, and furtherly generate sequences for image scrambling and encryption. Therefore, our encryption method is plaintext related so that it can effectively resist plaintext attacks.

The rest of the paper is organized as follows. The non-equilibrium chaotic system is introduced in Section 2. The ZigZag transform method and its modified algorithm are given in Section 3. The proposed image encryption and decryption scheme are introduced in Section 4. Section 5 presents the experimental results and the security of the algorithm. Finally, we conclude this paper in Section 6.

2 The Non-equilibrium Chaotic System

Chaotic system has been successfully applied in image encryption because of its unique characteristics. From the aspect of whether there exist equilibrium points, existing chaotic systems can be divided into the following types: with one or two stable equilibrium points, with planes or surfaces of equilibria, with non-hyperbolic equilibria, and with no equilibrium point. Among them, the non-equilibrium chaotic system has become a research hotspot because of its unexpected responses to perturbations. The characteristics of the non-equilibrium chaotic system make it more suitable for data encryption and information hiding. In Ref. [21], a non-equilibrium three-dimensional chaotic system is presented which is expressed by:

$$\begin{cases} \dot{x_1} = a\,\mathrm{sgn}(x_2) + bx_3 \\ \dot{x_2} = x_3 + d \\ \dot{x_3} = -cx_1 - x_3 \end{cases} \tag{1}$$

where $sgn(x)$ is symbolic function, x_1, x_2, x_3 are three state variables, and a, b, c, d are control parameters of the chaotic System (1). In [21], the authors have shown that when the system parameters of the chaotic system are set as $a = 2.8, b = 2.8, c = 1, d = 0.8$, System (1) exhibits a very complicated dynamics phenomenon and will produce chaos. The three-dimensional view of the chaotic strange attractor and some dynamical behavior in different planes for System (1) is shown in Figure 1.

It can be seen from Figure 1 that System (1) has strong chaotic behavior. In addition, it has more complex dynamic characteristics compared with low-dimensional chaotic systems.

3 Modified ZigZag Transform

ZigZag transform is a simple data scanning method. For a matrix, its basic idea is to scan the elements of a matrix from the upper left corner to the lower right corner in ZigZag order to disturb the data. Figure 2 shows the process of transforming a data matrix by using ZigZag transform.

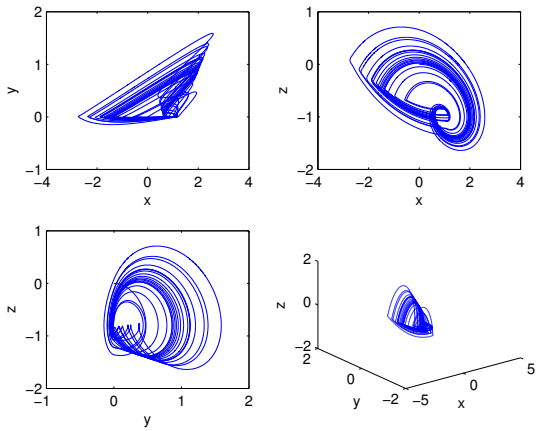


Figure 1: Typical dynamical behaviors of the non-equilibrium chaotic system

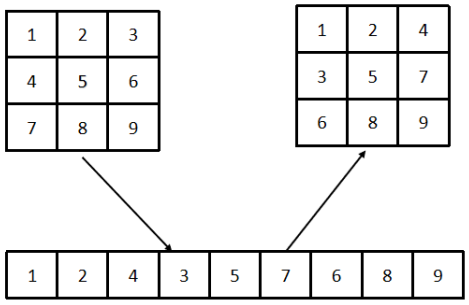


Figure 2: The flow chart of the ZigZag transform

From the results of the ZigZag transform, most locations of the elements in the original matrix have been changed, so that a certain scrambling effect can be achieved by the ZigZag transform. However, the positions of the first and last elements "1" and "9" are not changed after scrambling. In addition, ZigZag transform has periodicity. After a certain number of rounds of transform, the resulting matrix will become the original data matrix. To solve the problems mentioned above and increase the complexity of the ZigZag transform, we propose an improved ZigZag transform scheme in this paper. Compared with the traditional ZigZag transform method, the modified ZigZag transform method add a sequence transform step in the middle. The specific process of the modified ZigZag transform method is shown in Figure 3.

The Step 2 in Figure 3 is a sequence transform method and its technical details are as follows: The image vector is divided into three sub vectors with equal length, which are recorded as S1, S2 and S3 respectively. The chaotic sequence is used to generate a random number sequence, whose elements can only be 1, 2 or 3. If the random number is 1, the original sequence will be converted to S2, S3, S1; If the random number is 2, the original sequence will be converted to S3, S1, S2; If the random number is 3, the original sequence will be changed to be S3, S2, S1. All the three transforms can change not only the head and tail elements but also the sequence structure of the original sequence, so as to destroy the periodicity of the basic ZigZag transform. As a result, The improved ZigZag transform

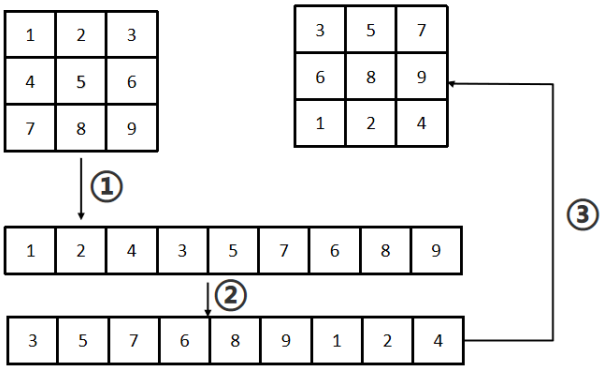


Figure 3: The flow chart of the modified ZigZag transform

scheme can not only overcome the shortcomings of the basic ZigZag transform, but also increase the complexity of the ZigZag transform, and make the scrambling process more difficult to be attacked.

4 Image Encryption Algorithm

4.1 Calculating The Initial Values of The Chaotic System

In this paper, the SHA256 hash function of the plain image is utilized to produce the initial values of the chaotic system. SHA 256 is a cryptographic hash function, its hash value is a hexadecimal number sequence with 64 digits long. If the input message is slightly different, it will output completely different results. For the image to be encrypted, the calculated SHA 256 hash function value can be expressed as $q = q_1 q_2 ... q_{64}$. The initial values x_0, y_0 and z_0 for the chaotic System (1) is calculated as follows:

$$\begin{cases} x_0 = \frac{1}{100}\sum_{i=1}^{20} q_i \\ y_0 = \frac{1}{100}\sum_{i=21}^{40} q_i \\ z_0 = \frac{1}{100}\sum_{i=41}^{64} q_i \end{cases} \tag{2}$$

4.2 The Encryption Method

Suppose the size of the plain image P is $M \times N$, where M and N represent the height and width of the image respectively. The main steps of the image encryption algorithm are as follows:

Step 1: Choose the system control parameters a, b, c, d of the no-equilibrium three-dimensional chaotic System (1).

Step 2: Iterate the chaotic System (1) for $L+2000$ times with the initial values x_0, y_0, z_0, remove the former 2000 values and then we can obtain three chaotic sequences x_s, y_s, z_s of length L, where $L = M \times N$.

Calculate two sequences F, S with x_s, y_s, z_s by

$$\begin{cases} u = x_s + y_s - z_s \\ F = (|u| \times 10^{15} \mod 3) + 1 \\ S = |u| \times 10^{15} \mod 256 \end{cases} \tag{3}$$

Set $t = 1$.

Step 3: Transform the image matrix P into an one-dimensional pixel vector P_V by ZigZag order. Set $S_1 = P_V(1, 1\colon T)$, $S_2 = P_V(1, T+1\colon 2T)$, $S_3 = P_V(1, 2T+1\colon 3T)$, where $T = L/3$.

Step 4: If $F(t) = 1$, Set $P_V = [S2, S3, S1]$; if $F(t) = 2$, Set $P_V = [S3, S1, S2]$; else if $F(t) = 3$, Set $P_V = [S3, S2, S1]$.

Step 5: Transform the pixel vector P_V back into image matrix P by row-major order.

Set $t = t+1$. Step(3)-Step(5) is repeated until $t > K$ so that the image is fully scrambled, where K is the number of scanning rounds.

Step 6: Transform the scrambled image P into an one-dimensional pixel vector P_V by row priority order.

Step 7: Perform the xor operation on P_V using the random sequences S:

$$\begin{cases} C_V(1, i) = P_V(1, i) \oplus S(1, i) \end{cases} \tag{4}$$

where $i = 2, 3, \cdots, L$ and symbol " $\oplus$ " is the bitwise exclusive or operator and C_V is the ciphertext vector.

Step 8: Convert C_V into encrypted gray image C.

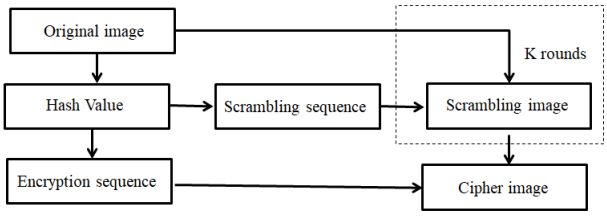


Figure 4: The flow chart of the encryption algorithm

The flow chart of the proposed encryption algorithm is shown in Figure 4.

4.3 The Decryption Method

The decryption process is the inverse process of the encryption process which mainly contains the following steps:

Step 1: Transform the cipher image C into one-dimensional pixel vector C_V by row priority order.

Step 2: Calculate the scrambled image vector P_V as follows:

$$\begin{cases} P_V(1,i) = C_V(1,i) \oplus S(1,i) \end{cases} \quad (5)$$

where $i = 2, 3, \cdots, L$. Convert P_V back into image matrix P, set $t = K$.

Step 3: Transform the image matrix P into an one-dimensional pixel vector P_V by row priority. Set $S_1 = P_V(1\colon T)$, $S_2 = P_V(T{+}1\colon 2T)$, $S_3 = P_V(2T{+}1\colon 3T)$.

Step 4: If $F(t) = 1$, Set $P_V = [S3, S1, S2]$; if $F(t) = 2$, Set $P_V = [S2, S3, S1]$; else if $F(t) = 3$, Set $P_V = [S3, S2, S1]$.

Step 5: Transform the pixel vector P_V back into matrix P by inverse ZigZag transform.

Set $t = t - 1$. Repeat Step(3)-Step(5) K rounds.

Step 6: Convert matrix P into plain image.

5 Test and Analysis of the Proposed Scheme

The Matlab software is used as an experimental platform for experiments. Three images i.e. plaint, house and peppers (240×240) are taken for testing. The system parameters of the chaotic system are set as $a = 2.8, b = 2.8, c = 1, d = 0.8$ and the scanning parameter K is set to be 10. In the following subsections, the experimental results and several different security analyses are given.

5.1 The Encrypted Image

The proposed algorithm is used to encrypt and decrypt the three images. The experimental results are shown in Figure 5. As can be seen from Figure 5, the correlations between the encrypted images and the original images are very small, so the content in the original images can be effectively protected. In addition, we can also find that the decrypted image is exactly the same as the original image, which proves that the proposed algorithm can decrypt the original image correctly.

5.2 Key Space Analysis

A good encryption algorithm should have enough keyspace. We use the SHA256 algorithm to process the image to obtain a 256-bit key string. Furtherly, the scrambling and diffusion key parameters are generated based on the key string. The secret key of the proposed algorithm is comprised of the chaotic system parameters a, b, c, the chaotic system initial values x_0, y_0, z_0 and the number of scanning rounds K. Suppose the precision of the system parameters is 10^{15}, the keyspace of the proposed algorithm will be more than 10^{90}, which can effectively resist exhaustive attacks.

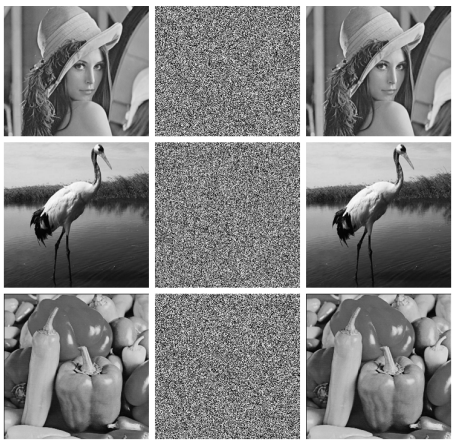


Figure 5: The experimental results of the encrypted image. (Left) The plain image. (Middle) The ciphered image. (Right) The decrypted image.

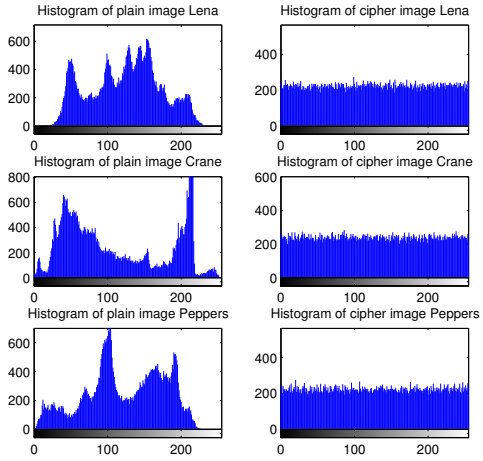


Figure 6: Histograms of plain images and cipher images

5.3 Histogram Analysis

The image grey histogram can be obtained by counting the grey value of each pixel information of the image. A good encrypted image needs a uniformly distributed histogram to resist statistical analysis. The histogram of the plain images and ciphered images is show in Figure 6. It can be seen from Figure 6, the pixel distribution of the original images is extremely uneven, while the pixel distribution of the cipher images is very uniform so that they can resist the histogram attacks.

5.4 Correlation Analysis

The adjacent pixels of the image have strong correlation. A good encryption algorithm needs to eliminate the correlation in the horizontal, vertical and diagonal directions of the image. The correlation of adjacent pixels can

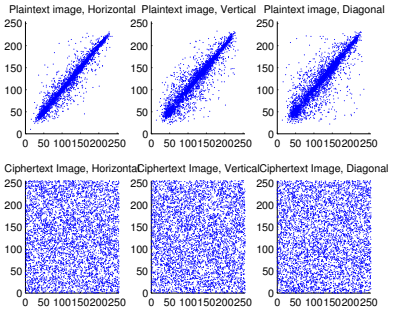


Figure 7: Correlation distributions of plain image Plant and cipher image Plant in each direction

be calculated by [17]:

$$r_{xy} = \frac{\sum_{i=1}^{N}((x_i - E(x))(y_i - E(y)))}{\sqrt{(\sum_{i=1}^{N}(x_i - E(x))^2)(\sum_{i=1}^{N}(y_i - E(y))^2)}} \quad (6)$$

where $E(x) = \sum_{i=1}^{N} x_i$, $E(y) = \sum_{i=1}^{N} y_i$, x_i and y_i are grey-level values of the selected adjacent pixels, and N is the number of sample pixels.

Table 1: Correlation coefficients of the plain images and cipher images

Images		Horizontal	Vertical	Diagonal
Lena	Plain	0.9409	0.9695	0.9101
	Cipher	-0.0045	-0.0167	0.0213
Crane	Plain	0.9810	0.9842	0.9697
	Cipher	-0.0082	-0.0123	-0.0079
Peppers	Plain	0.9462	0.9554	0.9015
	Cipher	-0.0251	-0.0093	-0.0039

5000 pixels and its adjacent pixels in three directions of the plaint images and cipher images are randomly select respectively to test the correlation of adjacent pixels, and the results are given in Table 1. In addition, the correlation distribution of r_{xy} of the plain image is plotted in Figure 7.

It is easy to see from Table 1 and Figure 7 that the adjacent pixels of the plain images have strong correlation, while the correlation of cipher images is very close to 0. As a result, the proposed image encryption method can eliminate the correlation of adjacent pixels.

5.5 Information Entropy Analysis

Information entropy reflects the randomness of the image. If the information entropy of an image is greater, then its randomness is greater and its security is higher [2]. The information entropy is calculated as follows:

$$H(m) = -\sum_{i=0}^{255} P(m_i)\log_2 P(m_i) \quad (7)$$

where m_i denotes the ith grey level for the digital image and $P(m_i)$ is the probability of m_i.

The information entropy test results of the plain images and encrypted images are shown in Table 2. The information entropies of the encrypted images are all close to the ideal value 8, which proves that the encryption effect of the image obtained by the proposed method is ideal.

Table 2: The results of entropy analysis

Images		Information entropy
Lena	Plain	7.4539
	Cipher	7.9971
Crane	Plain	7.5102
	Cipher	7.9972
Peppers	Plain	7.5553
	Cipher	7.9968

5.6 Analysis of Differential Attack Resistance

An effective encryption algorithm should be sensitive to plain images. In other words, a small change in the plain image should lead to a huge change in the cipher image to resist the differential attack. The number of pixels change rate (NPCR) and the unified averaged changed intensity (UACI) [18] are two common differential attack analysis methods. Suppose there is only one-pixel difference between two plain images. The formulas for calculating NPCR and UACI are

$$NPCR = \frac{\sum_{ij} D_{ij}}{W \times H} \times 100\% \quad (8)$$

$$UACI = \frac{1}{W \times H}\frac{\sum_{ij}(C_1(i,j) - C_2(i,j))}{255} \times 100\% \quad (9)$$

where C_1 and C_2 are the encrypted images for the plain images and D_{ij} is defined by

$$D_{ij} = \begin{cases} 0 & \text{if } C_1(i,j) = C_2(i,j) \\ 1 & \text{if } C_1(i,j) \neq C_2(i,j) \end{cases} \quad (10)$$

Table 3: NPCR and UACI

Images	NPCR(%)	UACI(%)
Lena	99.5972	33.5519
Crane	99.6386	33.3237
Peppers	99.6111	33.4687

The calculation results of NPCR and UACI is shown in Tabel 3. The ideal values of NPCR and UACI are 99.6% and 33.4% respectively. It can be found that the calculation results are very close to the ideal values, so the algorithm in this paper has a good ability to resist differential attacks.

Table 4: Performance comparison with other methods

Index	Ref. [19]	Ref. [23]	Proposed
Horizontal Correlation	-0.0230	0.0039	-0.0045
Vertical Correlation	0.0019	-0.0314	-0.0167
Diagonal Correlation	-0.0034	0.0158	0.0213
NPCR(%)	99.6200	99.6185	99.5972
UACI(%)	33.5100	28.7344	33.5519
Information entropy	7.9974	7.9890	7.9971

5.7 Performance Comparison with Other Methods

To further show the effectiveness of the proposed method, the proposed method is compared with the other two image encryption methods proposed in Ref. [19] and Ref. [23] from the aspects of entropy, NPCR, UACI, and correlation analysis for the Lena image. The specific results are given in Table 4.

From Table 4 we can see that the performance of the proposed method is competitive compared with Ref. [19] and Ref. [23]. Besides, the proposed method is relatively easy to implement so that it is suit for image encryption.

6 Conclusions

The ZigZag transform method is a classical image scrambling method and is often used for image encryption. The main drawbacks of the ZigZag transform method are studied and a modified ZigZag transform method is proposed to overcome the shortcomings of the ZigZag transform method. Further, the modified ZigZag transform algorithm is used for image scrambling and encryption. Experimentation is done on three classical images and the results of the encryption and decryption test and security analysis show that the proposed method is of practical significance in image encryption.

Acknowledgments

This work is partially supported by the National Natural Science Foundation of China (No.11871417). The authors gratefully acknowledge the anonymous reviewers for their valuable comments.

References

[1] S. Ahadpour, Y. Sadra, "A chaos-based image encryption scheme using chaotic coupled map lattices," *International Journal of Computer Applications*, vol. 49, no. 2, pp. 15-18, 2012.

[2] R. E. Boriga, A. C. Dascalescu, A. V. Diaconu, "A new fast image encryption scheme based on 2D chaotic maps, *IAENG International Journal of Computer Science*, vol. 41, no. 4, pp. 249-258, 2014.

[3] J. C. Dagadu, J. Li, E. O. Aboagye, F. K. Deynu, "Medical image encryption scheme based on arnold transformation and ID-AK protocol," *International Journal of Network Security*, vol. 19, no. 5, pp. 776-784, 2017.

[4] J. C. Dagadu, J. Li, E. O. Aboagye, F. K. Deynu, "Medical image encryption scheme based on multiple chaos and DNA coding," *International Journal of Network Security*, vol. 21, no. 1, pp. 83-C90, 2019.

[5] H. Dong, E. Bai, X.Q. Jiang, "Color image compression-encryption using fractional-order hyperchaotic system and DNA coding," *IEEE Access*, vol. 8, pp. 163524-163540, 2020.

[6] H. Gao, X. Y. Wang, "An image encryption algorithm based on dynamic row scrambling and ZigZag transform," *Chaos, Solitons and Fractals*, vol. 147, no. 6, 110962, 2021.

[7] M. Ghebleh, A. Kanso, D. Stevanovi, "A novel image encryption algorithm based on piecewise linear chaotic maps and least squares approximation," *Multimedia Tools and Applications*, vol. 77, no. 6, pp. 7305-7326, 2018.

[8] H. Huang, D. Cheng, "3-image bit-level encryption algorithm based on 3d nonequilateral arnold transform and hyperchaotic system," *Security and Communication Networks*, vol. 7, pp. 1-13, 2020.

[9] P. Li, J. Xu; J. Mou. F. F. Yang, "Fractional-order 4D hyperchaotic memristive system and application in color image encryption," *EURASIP Journal on Image and Video Processing*, vol. 26, no. 10, pp. 11-23, 2017.

[10] Y. B. Mao, G. R. Chen, S. G. Lian, "A novel fast image encryption scheme based on 3D chaotic Baker maps," *International Journal of Bifurcation & Chaos*, vol. 14, no. 10, pp. 3613-3624, 2004.

[11] R. Matthews, "On the derivation of a chaotic encryption algorithm," *Cryptologia*, vol. 13, no. 1, pp. 29-C41, 1989.

[12] M. Naim, A.A. Pacha, C. Serief, "A novel satellite image encryption algorithm based on hyperchaotic systems and Josephus Problem," *Progress in space research*, vol. 67, no. 7, pp. 2077-2103, 2021.

[13] H. Natiq, N. M. G. Al-Saidi, M. R. M. SaidAdem Kilicman, " A new hyperchaotic map and its application for image encryption," *The European Physical Journal Plus*, vol. 133, no. 6, pp. 5-18, 2018.

[14] G. Qu, X. Meng, Y. Yin, "Optical color image encryption based on Hadamard single-pixel imaging and

Arnold transform," *Optics and Lasers in Engineering*, vol. 137, no. 20, 106392, 2021.
[15] X. Wang, Y. Su, "Image encryption based on compressed sensing and DNA encoding," *Signal Processing Image Communication*, vol. 12, 116246, 2021.
[16] J. H. Wu, X. F. Liao, B. Yang, "Image encryption using 2D Hnon-Sine map and DNA approach," *Signal Processing*, vol. 153, no. 12, pp. 11-23, 2018.
[17] X. Wu, H. Kan, and J. Kurths, "A new color image encryption scheme based on DNA sequences and multiple improved 1D chaotic maps," *Applied Soft Computing*, vol. 37, pp. 24-39, Dec. 2015.'
[18] Y. Wu, J. P. Noonan, S. Agaian, "NPCR and UACI randomness tests for image encryption," *Journal of Selected Areas in Telecommunications*, vol. 1, no. 2, pp. 31-38, 2011.
[19] L. Xu, Z. Li, J. Li, W. Hua, "A novel bit-level image encryption algorithm based on chaotic maps," *Applied Soft Computing*, vol. 78, no. 3, pp. 17-25, 2016.
[20] G. D. Ye, "A chaotic image encryption algorithm based on information entropy," *International Journal of Bifurcation and Chaos*, vol. 28, no. 1, pp. 1-11, 2018.
[21] S. Zhang, X. Wang, Z. Zeng, "A simple no-equilibrium chaotic system with only one signum function for generating multidirectional variable hidden attractors and its hardware implementation," *Chaos*, vol. 30, no. 5, 53129, 2020.
[22] X. Zhang, L. Wang, Y. Wang, Y. Niu, Y. Li, "An image encryption algorithm based on hyperchaotic system and variable-step josephus problem," *International Journal of Optics*, vol. 4, pp. 1-15, 2020.
[23] X. Zhang, Z. Zhou, Y. Niu, "An image encryption method based on the Feistel network and dynamic DNA encoding," *IEEE Photonics Journal*, vol. 10, no. 4, pp. 1-14, 2018.

Biography

Chunming Xu is an associate professor at the mathematics and statistical from Yancheng Teachers University, PR China. His main research interests include image processing and artificial intelligence.

Yong Zhang is a professor at the mathematics and statistical from Yancheng Teachers University, PR China. His main research interests include combinatorics and optimization.

Privacy-Preserving Scoring Mechanism

Zhuliang Jia[1], Xueling Zhao[1], and Jiahao Pan[2]
(Corresponding author: Zhuliang Jia)
School of Computer Science, Shaanxi Normal University, Xi'an 710119 China[1]
School of Mathematics and Statistics, Shaanxi Normal University, Xi'an 710119 China[2]
Email: zhuliang@snnu.edu.cn
(Received Mar. 20, 2022; Revised and Accepted Oct. 11, 2022; First Online Oct. 15, 2022)

Abstract

Secure multi-party computation(SMC) is a hot topic in the international cryptographic field. The scoring mechanism is an effective evaluation mechanism to select high-quality objects from similar candidates and is also used in various competitions. There is currently no scheme that can ensure the security of judges' private information while following the actual scoring rules of the competition. To solve this problem, we propose a privacy-preserving scoring mechanism that protects the judges' information and does not disclose any information except the final score.

Keywords: Privacy-preserving; Scoring Mechanism; Secure Multi-party Computation

1 Introduction

It is unrealistic to place unconditional trust in judges and players, as they may pervert the fair scoring process for illegal gain. Therefore, it is important to design a secure scoring system that does not reveal each judge's score. This paper adopts the way of Secure multi-party computation [14] to design the scoring mechanism protecting the judges' privacy.

A privacy-preserving scoring mechanism should not only protect judges' privacy, but also follow certain scoring rules. We use the most common scoring rules in various competitions: (1) Remove a Min score and a Max score; (2) Remove two highest scores and the two lowest scores. Although there has been a lot of research about other secure computation [1, 4, 6–8, 10–13], as far as we know, there is no research about privacy-preserving scoring mechanism. The main contributions of this paper are as follows:

1) We proposed and studied a new problem in SMC, that is, privacy-preserving scoring mechanism, and the solution to this problem is proposed.

2) Based on the Lifted ElGamal encryption system, we design a secure scoring protocol. The score and the ranking of score are encrypted by different public keys. The decryption of ranking information can not be associated with the judge's identity and score.

2 Preliminaries

2.1 Semi-honest Models and Security

Definition 1. *The model in which all participants are semi-honest is called a semi-honest model [3]. Suppose that a multi-party protocol π respectively asked $P_i(i \in [1,n])$ to execute the function $f(X)$, where x_i is the input of $P_i(i \in [1,n])$, $X = (x_1, \cdots, x_n)$. Let $view_I^\pi(X) = (I, view_s^\pi(X), \cdots, view_t^\pi(X), f_I(X))$ is the view of P_i during complete π, where r_i is randomness of P_i and $m_i^j(j \in [1,k])$ is the $j-th$ message received from others. We can say that π is secure against semi-honest adversaries compute function $f(X)$, for any participants $I = \{P_s, \cdots, P_t\} \subset \{P_1, \cdots, P_n\}(1 \le s < t \le n)$, if there exist probabilistic polynomial time simulator S such that:*

$$\{S(I, (x_s, \cdots, x_t), f_I(X))\}_X \stackrel{c}{\equiv} \{view_I^\pi(X)\}_X \quad (1)$$

where $\stackrel{c}{\equiv}$ denote computationally indistinguishable.

The above method of constructing simulators S to prove protocol security is called the simulation paradigm [9]. This method is used to prove the security of all protocols in this paper.

2.2 Lifted-ElGamal Threshold System

Definition 2. *Lifted-ElGamal threshold system [5] based on the ElGamal cryptosystem [2], and the threshold is constructed by replacing the m with g^m. The specific structing process is as follows:*

- *Init(κ): Let p is a κ-bit big prime and g be a generator of Z_p^*. P_i select k_i as personal private key, compute and share personal public key$h_i = g^{k_i} \bmod p$. Then, all participants compute public key as $h = \Pi_{i=1}^n h_i \bmod p = g^{\Sigma_{i=1}^n k_i} \bmod p$.*
- *Enc(m): For a plaintext $m \in Z_p$, we select a random number $r \in Z_p^*$ to generate ciphertext as $C = (C_1, C_2) = (g^r \bmod p, g^m h^r \bmod p)$.*

- *Dec(C): Additive Homomorphism: For a ciphertext* $C = (C_1, C_2)$*, all participants jointly decrypt as* $d = \frac{C_2}{\Pi_{i=1}^{n} C_1^{k_i}} \ mod\ p$.
- *For* $m_1, m_2 \in Z_n$*, we can find:*

$$E(m_1)E(m_2) = E(m_1 + m_2 \ mod\ p).$$

3 Problem Statement

3.1 Mechanism Model

In some competitions, all judges $P_1, \cdots, P_n$ give grade $x_1, \cdots, x_n \in U = \{0, 1, 2, \cdots, m\}$ for a contestant's performance. Let's $x = \{x_1, \cdots, x_n\}$. The scoring rules are: remove the two highest scores x_{max_1}, x_{max_2}, remove the two lowest scores x_{min_1}, x_{min_2}, the sum of the remaining scores $f(x) = (\sum_{i=1}^{n} x_i) - (x_{max_1} + x_{max_2} + x_{min_1} + x_{min_1})$ as the final score of the contestant.

3.2 Design Goals

Our design goal is to design secure and efficient scoring mechanisms. Our protocol is required to be secure under a semi-honest model.

4 Secure Scoring Mechanism

4.1 Scheme and Design

How to solve the problem under model ? We can solve the problem in this way. At first, $x_i (i \in [1, n])$ is securely sorted in ascending order, and the serial number is denoted as sn_i. Obviously, the serial number of two lowest scores and the two highest scores correspond to $1, 2, n, n-1$, respectively. Then, the final score f can be further calculated. We will explain details as follows:

- The judges $P_i (i \in [1, n])$ respectively construct row vector $\overrightarrow{x_i} = (x_{i0}, \cdots, x_{im})$ according x_i as follows, where $j \in [0, m]$:

$$x_{ij} = \begin{cases} 1, & j = x_i \\ 0, & else \end{cases},$$

- Matrix X can be formed from vectors $\overrightarrow{x_1}, \cdots, \overrightarrow{x_n}$,

$$X = \begin{bmatrix} \overrightarrow{x_1} \\ \vdots \\ \overrightarrow{x_n} \end{bmatrix} = \begin{bmatrix} x_{10} & \cdots & x_{1m} \\ \vdots & \ddots & \vdots \\ x_{n0} & \cdots & x_{nm} \end{bmatrix}.$$

- After initializing $d = \emptyset$, P_i does the following, respectively:
  - P_i generate ns_i according x_i and matrix X:

$$sn_i = \begin{cases} \sum\limits_{j=0}^{x_i - 1} \sum\limits_{k=1}^{n} x_{kj} + \sum\limits_{j=x_i, k \le i} x_{kj}, & x_i > 0 \\ \sum\limits_{k \le i} x_{k0}, & x_i = 0 \end{cases}.$$

  - P_i computes and updates set $d \leftarrow d \cup \{\langle x_i, sn_i \rangle\}$.
  - P_i sends the updated set d to P_{i+1}, until the work of P_n is finished.
- P_n makes $A = \{3, 4, \cdots, n-2\}$. It is easy to find the following conclusions.

Proposition 1. *After removing the two largest and the two smallest numbers, the sum of the remaining elements of set* X *is* $f = \sum_{sn_i \in A} x_i$.

4.2 Protocol 1: Secure Scoring Protocol

Input: $P_1, \cdots, P_n$ input $x_1, \cdots, x_n$; m

Output: f

Setup: Referees run the Lifted ElGamal threshold cryptosystem twice, keep their private keys sk_i', sk_i'' respectively, publish the jointly generated public keys pk', pk'', and initialize set $D = \emptyset$.

- For $i \in [1, n]$, P_i firstly computes the row vector $\overrightarrow{x_i} = (x_{i0}, \cdots, x_{im})$ according to the score x_i as follows:

$$x_{ij} = \begin{cases} 1, & j = x_i \\ 0, & else \end{cases},$$

where $j \in [0, m]$. Then, P_i encrypts the vector $\overrightarrow{x_i}$ with pk'' to $C_i = (E''(x_{i0}), \cdots, E''(x_{im}))$, and publish C_i. Finally, we can get $C = [c_1 \cdots c_n]^T$.

- For $i \in [1, n]$, the P_i does following work respectively:
  - P_i encrypts x_i with pk' to $X_i = E'(x_i)$.
  - P_i computes SN_i based on the score x_i and matrix C, where

$$SN_i = \begin{cases} (\prod\limits_{j=0}^{x_i - 1} \prod\limits_{k=1}^{n} c_{kj}) \cdot (\prod\limits_{j=x_i, k \le i} c_{kj}), & x_i > 0 \\ \prod\limits_{k \le i} c_{k0}, & x_i = 0 \end{cases}.$$

    Then, we can get $D_i = \langle SN_i, X_i \rangle$.
  - P_i updates D:
    - For $\iota \in [1, \cdots, i-1]$, here is: $X_\iota \leftarrow X_\iota \cdot E'(0), SN_\iota \leftarrow SN_\iota \cdot E''(0)$.
    - $D \leftarrow D \cup D_i$.
  - P_i sends the updated D to the next referee P_{i+1}, until $i + 1 = n$. Then, we can get $D = \{D_1, \cdots, D_n\}$.
- Referees jointly decrypt $SN_1, \cdots, SN_n$ to get $sn_1, \cdots, sn_n$. Compute $F = \prod_{sn_i \in \{3, 4, \cdots, n-2\}} X_i$.
- Judges jointly decrypt F to get f.

Note 1. *Using the similar principle, we can easily obtain a secure scoring protocol with the highest score and the lowest score removed.*

5 Security Analysis

5.1 Correctness

According to the additive homomorphism of Lifted ElGamal cryptosystem, if we record $c_{kj} = E(x_{kj})$, when $x_i > 0$, there is $X_{i,1} = (\prod_{j=0}^{x_i-1}\prod_{k=1}^{n} c_{kj}) \cdot (\prod_{j=x_i,k\leq i} c_{kj}) = E(\sum_{j=0}^{x_i-1}\sum_{k=1}^{n} x_{kj}) + (\sum_{j=x_i,k\leq i} x_{kj})$; when $x_i = 0$, there is $X_{i,1} = \prod_{k\leq i} c_{k0} = E(\sum_{k\leq i} x_{k0})$. Then, according to Proposition 1, we know that protocol 1 is correct.

5.2 Security

We construct the simulator S as follows:

- After S receives $(I, (x_2, \cdots, x_n), f_I(x_1, \cdots, x_n))$, it randomly select a x_1^* so that $f_I(x_1^*, x_2 \cdots, x_n) = f_I(x_1, \cdots, x_n)$ holds.
- S initializes the row vector $\overrightarrow{x_1}^* = (x_{10}^*, \cdots, x_{1m}^*)$ according to x_1^*. S encrypts $\overrightarrow{x_1}^*$ with pk'' to C_1^*.

  Similarly, for $i \in [2, n]$, S respectively initializes $\overrightarrow{x_i} = (x_{i0}, \cdots, x_{im})$ according to x_i. Then S encrypts $\overrightarrow{x_i}$ with pk'' to C_i. Finally, we can get C^*.
- Firstly, S initializes $D^* = \emptyset$ and encrypts x_1^* with pk' to $X_1^* = E'(x_1^*)$. Then, S gets ${SN_1}^*$ according to x_1^* and matrix C^*. In the end, S gets $D_1^* = \langle X_1^*, SN_1^* \rangle$.
- For $i \in [2, n]$, S does the following in sequence:
  - S encrypts x_i with pk' to $X_i^* = E'(x_i)$.
  - S computes SN_i^* based on x_i and C^*. In the end, S gets $D_i^* = \langle X_i^*, SN_i^* \rangle$.
  - S updates D^*.
- S decrypts $SN_1^*, \cdots, SN_n^*$ to get $sn_1^*, \cdots, sn_n^*$, and then computes $F^* = \prod_{sn_i \in \{3,4,\cdots,n-2\}} X_i^*$.
- S decrypts F^* to get f^*.

During Protocol 1, $view_I^{\pi}(x_1, \cdots, x_n) = \{(x_2, \cdots, x_n), C_1, \langle X_1, SN_1 \rangle, f_I(x_1, \cdots, x_n)\}$. Let

$$\begin{aligned} & S(I, (x_2, \cdots, x_n), f_I(x_1, \cdots, x_n)) \qquad (2) \\ = \ & \{(x_2, \cdots, x_n), C_1^*, \langle X_1^*, SN_1^* \rangle, f_I(x_1^*, x_2, \cdots, x_n)\}. \end{aligned}$$

Since I cannot decrypt C_1 and C_1^*, according to the semantic security of Lifted ElGamal threshold cryptosystem, $C_1^* \overset{c}{\equiv} C_1$, $X_1^* \overset{c}{\equiv} X_1$, $SN_1^* \overset{c}{\equiv} SN_1$. Apart from this, $f_I(x_1^*, x_2, \cdots, x_n) = f_I(x_1, \cdots, x_n)$, so we can say that, $\{view_I^{\pi}(x_1, \cdots, x_n)\} \overset{c}{\equiv} \{S(I, (x_2, \cdots, x_n), f_I(x_1, \cdots, x_n))\}$.

6 Performance Evaluation

6.1 Performance of Protocols

Computational Complexity. In Protocol 1, generating 2 pairs of public keys requires $2n$ modular exponential operations. The participants encrypt vector $\overrightarrow{x_i}(x_{ij} \in \{0, 1\})$ require $2mn$ times modular exponential operations. During the computation process, the participants need $3n$ modular exponential operations to encrypt x_i and $2n^2 - 2n$ modular exponential operations to update the set D. Decrypting data needs $n^2 + n$ modular exponential operations. Therefore, Protocol 1 requires total of $3n^2 + 2nm + 4n$ modular exponential operations.

Communication Complexity. In Protocol 1, it requires $n - 1$ communications to generate the key, requires $2(n - 1)$ communications in the computation process, and requires requires $2(n - 1)$ communications to jointly decrypt. Therefore, Protocol 1 requires total of $5(n - 1)$ communications.

The specific analysis results shown in Table 1 below.

Protocol Complexity

Complexity	**Protocol 1**
Computational	$3n^2 + 2nm + 4n$
Communication	$5(n - 1)$

6.2 Experiment

Experimental Environment. Windows10 64-bit operating system, the processor parameter is Intel(R) Core(TM) i5-9400 CPU@ 2.90GHz, 16.0GB memory.

Experimental Method. We use the control variable method to design experiments to investigate and study the relationship between protocol execution time and a single variable n, m when other influencing factors are fixed. In the experiment, we set the security parameter of the Lifted ElGamal threshold cryptosystem as 1024 bits, and recorded the average execution time required for 100 experiments. The specific experiments are as follows:

1) We test the relationship between protocol execution time and the number of participants n. In the actual experiment, we set $m = 50$, and the number of participants is $n = 5, 10 \cdots, 50$ to conduct the experiment. The experimental results are shown in Figure 1.

2) When other influencing factors are fixed, we test the relationship between protocol execution time and the data range m. In the actual experiment, we set $n = 10$, and the data range respectively take $m = 10, 20, \cdots, 100$ for the experiment. The experimental results are shown in Figure 2.

It can be seen from the experimental results that when other factors are fixed, the execution time

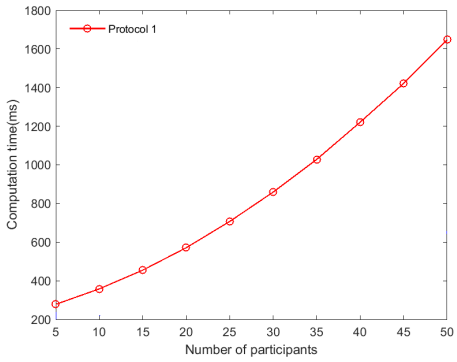


Figure 1: Time varies with the number of participants

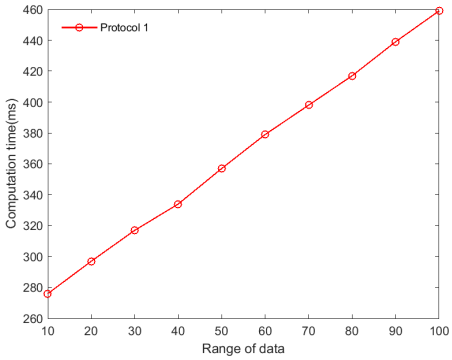


Figure 2: Time varies with the number of participants of protocol 1 roughly increases linearly with the increase of n^2, and the execution time of protocol 1 roughly increases linearly with the increase of m.

7 Conclusions

This paper proposes a new secure compute problem, that is, the secure scoring mechanism. This paper studies the privacy of two commonly used scoring mechanisms, one is to remove one lowest score and one highest score and then obtain the final score, the other is to remove the two lowest scores and the two highest scores and then obtain the final score. In the follow-up work, we will further study a more efficient scoring mechanism to protect referee privacy and a scoring mechanism to protect referee privacy under the malicious model.

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] T. Elgamal, "A public key cryptosystem and a signature scheme based on discrete logarithms," *IEEE Transactions on Information Theory*, vol. 31, no. 4, pp. 469-472, 1985.

[3] O. Goldreich, *The Fundamental of Cryptography: Basic Applications*, London: Cambridge University Press, pp.599-764, 2004.

[4] K. A. Kumar, A. Anjum, "A CHAOS maps based method using encryption scheme for securing DICOM images: A comparative analysis," *International Journal of Electronics and Information Engineering*, vol. 12, no. 3, pp. 128-135, 2020.

[5] J. Liu, N. Asokan, B. Pinkas, "Secure deduplication of encrypted data without additional independent servers," in *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, pp. 874-885, 2015.

[6] L. Liu, L. Wang, "Analysis of one identity-based integrity auditing and data sharing scheme," *International Journal of Electronics and Information Engineering*, vol. 12, no. 3, pp. 105-111, 2020.

[7] L. H. Liu, L. Wang, Z. Cao, *et al.*, "Algorithms for subset sum problem," *International Journal of Electronics and Information Engineering*, vol. 9, no. 2, pp. 106-114, 2018.

[8] W. R. Liu, X. He, Z. Y. Ji, *et al.*, "An improved authentication protocol for telecare medical information system," *International Journal of Electronics and Information Engineering*, vol. 12, no. 4, pp. 170-181, 2020.

[9] 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 & Developmental Disorders*, vol. 43, no. 9, pp. 2211-2217, 2013.

[10] S. Rezaei, M. A. Doostari, M. Bayat, "A lightweight and efficient data sharing scheme for cloud computing," *International Journal of Electronics and Information Engineering*, vol. 9, no. 2, pp. 115-13, 2018.

[11] S. Sciancalepore, R. D. Pietro, "PPRQ: Privacy-preserving MAX/MIN range queries in IoT networks," *IEEE Internet of Things Journal*, vol. 8, no. 6, pp. 5075-5092, 2021.

[12] H. Yang, J. Lee, M. S. Hwang, "A taxonomy of bluetooth security," *International Journal of Electronics and Information Engineering*, vol. 12, no. 2, pp. 43-65, 2020.

[13] Y. J. Yang, S. D. Li, R. M. Du, "Private maximum and minimum computation," *Journal of Cryptologic Research*, vol. 7, no. 4, pp. 483-497, 2020.

[14] A. C. Yao, "Protocols for secure computations," in *Proceedings of the 23th IEEE Symposium on Foundations of Computer Science*, pp. 160-164, 1982.

Biography

Zhuliang Jia 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.

Xueling Zhao 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.

Jiahao Pan is currently pursuing the M.S. degree with School of Mathematics and Information Science in Shaanxi Normal University. Her research interests focus on modern cryptography and applied mathematics.

Optimized Jacobian-based Saliency Maps Attacks

Wenwen Zhang[1], Xiaolin Zhang[1], Kun Hao[2], Jingyu Wang[1], and Shuai Zhang[1]
(Corresponding author: Xiaolin Zhang)
School of Information Engineering, Inner Mongolia University of Science and Technology, Baotou 014010, China[1]
School of Computer Science and Engineering, Northeastern University, Shenyang 110004, China[2]
Email: zhangxl6161@163.com
(Received Mar. 25, 2022; Revised and Accepted Oct. 12, 2022; First Online Oct. 15, 2022)

Abstract

Deep neural networks (DNNs) are vulnerable to malicious attacks by adversarial examples, including those modifying a small fraction of the input features named L_0 attacks. Jacobian-based saliency map attack (JSMA) is an effective and fast L_0 attack for fooling classification models, such as DNNS for image classification tasks. In this paper, we propose an improved version of JSMA, referred to as the optimized JSMA (OJSMA). We more accurately select and update the input features to obtain a more robust L_0 attack algorithm by the adversarial saliency map, which combines increasing and decreasing features and the objective function, which optimizes the L_2 norm of perturbations of selected features. Furthermore, we apply the OJSMA algorithm to an ensemble of models to further improve the transferability of the adversarial examples. Our experiments evaluated the performance of the adversarial examples generated by the OJSMA using the MNIST and CIFAR-10 datasets in four DNNs and four ensembles of models. The results indicate that the proposed method has a higher attack success rate, less distortion, and greater invisibility than the JSMA. Experiments also demonstrate, in some cases, very competitive results of our attack compared with other targeted JSMA variants, with significantly more minor adversarial perturbations.

Keywords: Adversarial Attacks; Deep Neural Networks; Ensembled of Models; Jacobian-based Saliency Maps

1 Introduction

The recent success of deep neural networks (DNNs) has dramatically improved performance on various vision tasks, including image classification [19] and object detection [5]. However, DNNs can be tricked by adding small perturbations to the input images. These intentionally crafted images are known as adversarial examples. Adversarial attacks are a process of using adversarial samples to make wrong network predictions.

Since the concept of adversarial examples was first introduced by Szegedy *et al.* [25] in 2014, several classical attack algorithms have emerged in the field of adversarial attacks. Adversarial attacks can be classified as L_0 , L_2 and L_∞ attacks according to distance metrics. Specifically, L_0 attacks use the L_0 norm to limit the number of modified features by changing only a few features and adding significant adversarial perturbations. Examples include Jacobian-based saliency map attack (JSMA) [18] and One Pixel attacks [22]. L_2 attacks use the L_2 norm to limit the root-mean square of the overall image perturbations by changing numerous features and adding insignificant adversarial perturbations. Examples include C&W attacks [2], zeroth-order optimization [3], DeepFool [15], and expectation over transformation [1]. Similarly, L_∞ attacks use the L_∞ norm to limit the modification size of individual features focusing on regions of the original image where the image grayscale dramatically changes and are closely related to the image gradient. Examples include the fast gradient sign method [7], basic iterative method [11], and projected gradient descent [13].

A DNN models a multidimensional function $F : X \to Y$, where X is an original sample, and Y is an original output label. We construct an adversarial example X^* from a benign sample X by adding a perturbation δ, which solves the following optimization problem:

$$\begin{aligned} & t = F(x+\delta) \\ & s.t.\, t \neq Y and \|\delta\|_p \leq \varepsilon \end{aligned} \tag{1}$$

where $X^* = X + \delta$ is an adversarial example, and t is the desired adversarial output. $\|\delta\|$ is the p-norm of the perturbation, which measures the size of the added perturbation, and ε is the upper limit of the perturbation.

In this paper, we focus on L_0 attacks [2, 18], called sparse attacks [8,17], which can accurately and efficiently fool DNNs by modifying only a small fraction of input features. This reveals very disturbing and astonishing properties of neural networks as it is possible to fool them by modifying few pixels [14, 22]. JSMA [18] is a widely used L_0 attack algorithm, which achieved a 97% adversarial success rate by modifying 4.02% input features per sample on average. The algorithm produces adversarial examples by constructing a mapping of input features to an output vector. We work on optimizing the JSMA algo-

rithm to obtain more robust adversarial examples. This paper makes the following contributions:

1) We introduce a variant of JSMA called optimized JSMA (OJSMA). OJSMA more accurately select and update the input features by the adversarial saliency map which combines increasing and decreasing features and the objective function which optimizes the L_2 norm of perturbations of selected features. We conducted multiple comparison experiments on two different datasets, and our algorithm has a higher attack success rate, produces fewer changes on the image and further enhances the invisibility of the perturbations compared to JSMA.

2) We compared OJSMA with other targeted L_0 attacks, including other JSMA variants (WJSMA, TJSMA) and C&W L_0 attacks. The results show that our method is comparable to other excellent variants in terms of attack success rate, but significantly smaller adversarial perturbations, and faster than C&W L_0 attack.

3) We apply the OJSMA algorithm to the ensemble of networks to further improve the transferability of the adversarial examples.

The rest of the article is organized as follows. Section 2 introduces the work related to our approach. Section 3 introduces our algorithm OJSMA. In Section 4, the experimental results are presented and analyzed and evaluated. Finally, in Section 5, conclude and discuss future work.

2 Related Works

2.1 JSMA

Papernot *et al.* [18] designed an efficient adversarial attack method in 2016, called JSMA. The algorithm selects features with the greatest impact on the output target class, called target features, using forward derivative and adversarial saliency maps [21], and then obtains adversarial examples X^* by modifying target features. If the adversarial example is unsuccessful in fooling the target DNN, the adversarial saliency map of the image X^* is calculated again until the attack is successful. The JSMA algorithm defined two adversarial saliency maps to select the target features:

$$S^+(X,t)[i] = \begin{cases} 0, \text{ if } \frac{\partial F_t(X)}{\partial X_i} < 0 \text{ or } \sum\limits_{j \neq t} \frac{\partial F_j(X)}{\partial X_i} > 0 \\ (\frac{\partial F_t(X)}{\partial X_i})|\sum\limits_{j \neq t} \frac{\partial F_j(X)}{\partial X_i}|, \text{ otherwise} \end{cases} \quad (2)$$

$$S^-(X,t)[i] = \begin{cases} 0, \text{ if } \frac{\partial F_t(X)}{\partial X_i} > 0 \text{ or } \sum\limits_{j \neq t} \frac{\partial F_j(X)}{\partial X_i} < 0 \\ |\frac{\partial F_t(X)}{\partial X_i}|(\sum\limits_{j \neq t} \frac{\partial F_j(X)}{\partial X_i}), \text{ otherwise} \end{cases} \quad (3)$$

where t is the target class, $\frac{\partial F_t(X)}{\partial X_i}$ is the forward derivative of the target class, and $\sum\limits_{j \neq t} \frac{\partial F_j(X)}{\partial X_i}$ is the sum of the forward derivative of the non-target classes. The adversarial saliency map of Equation (2) and Equation (3) generates adversarial examples by increasing and decreasing image features, respectively.

When input image features are increased to generate adversarial examples, features whose forward derivative for the target class and sum of forwarding derivative for the nontarget class is greater and less than zero, respectively, are chosen and assigned values to the corresponding features of the saliency map, as presented in Equation (2). Similarly, when decreasing input features of images to generate adversarial examples, features whose forward derivative for the target class are less than zero and the sum of forward derivative for the nontarget class is greater than zero are selected and assigned values of corresponding features of the saliency map, as presented in Equation (3). Then, the features with the highest value in the adversarial saliency map are selected and increased or reduced to their maximum or minimum values, respectively. This approach increases and decreases the output probabilities of the target class and other classes, respectively.

2.2 Transferability of Adversarial Examples

The transferability of adversarial examples allows an adversarial example generated for a specific known network to cause an unknown network to produce incorrect predictions. Current adversarial attack methods, while having high attack success rates under white-box conditions, are poor in terms of transferability. This issue is attributed to the fact that the adversarial samples generated using such methods are highly coupled to the structure and parameters of the model, and their perturbations are difficult to effectively attack against other models with different structures and parameters.

Previously, Xie *et al.* [30] improved the transferability of adversarial examples by creating diverse input patterns. The authors in [16] used an evolutionary algorithm to obtain target features and add perturbations at the target feature locations to obtain adversarial samples with good transferability. Meanwhile, [29] showed that image transformations can defend against adversarial examples under certain situations, which indicates adversarial examples cannot generalize well under different transformations. In this paper, we used the ensembled network as the target network without changing the attack algorithm to generate adversarial examples with better transferability.

2.3 Ensembled Network

The ensembled network [12, 26, 27] is a more robust network formed by ensembled multiple networks simultaneously in a certain way. Liu *et al.* [12] suggested that if an

adversarial image remains adversarial for multiple networks, it can transfer to other networks. Therefore, this strategy can be employed to improve the transferability of the adversarial image generated using our method. We follow the ensemble strategy proposed in [6], which fuses the logit activations to attack multiple networks simultaneously. Specifically, to attack an ensemble of K models, the logits are fused by the following:

$$l(x) = \sum_{k=1}^{K} \omega_k l_k(x) \tag{4}$$

where $l_k(x)$ is the logits output of the k-th model, and ω_k is the ensemble weight with $\omega_k \geq 0 \, and \sum_{k=1}^{K} = 1$.

3 Methodology

In this section, we present our OJSMA attack algorithm. Our algorithm targets the network ensembled by multiple DNNs, which is a targeted attack in a white-box context. The method can be generalized to any feedforward DNNs. The process of the OJSMA algorithm for producing a confrontation sample is as follows: 1) calculate the forward derivative of the target network, 2) design adversarial saliency maps, and 3) optimize perturbations and modify samples. Algorithm 1 shows our process for constructing adversarial examples. Figure 1 is the flow chart of our algorithm.

Algorithm 1 Crafting adversarial examples

Input:
Benign image X;
Target network output t;
Function learned by the network during training F;
Maximum distortion γ;
The change made to pixels θ;
Input image size K;

Output:
Adversarial examples X^*

Begin
$X^* \leftarrow X$, $\tau = \{1...|X|\}$, $max\_iter = \lfloor \frac{k \cdot \gamma}{2 \cdot 100} \rfloor$
while $F(X^*) \neq t$ and $iter < max\_iter$ and $\tau \neq \bigcirc$ **do**
  Compute forward derivative $\bigtriangledown F(X^*)$
  $p_1, p_2 =$ saliency_map$(\bigtriangledown F(X^*, \tau, t))$
  $\theta =$ compute_perturbation(X, t, p_1, p_2)
  modify p_1 and p_2 in X^* by θ
  Remove p_1, p_2 from τ
  $iter++$
end while
End

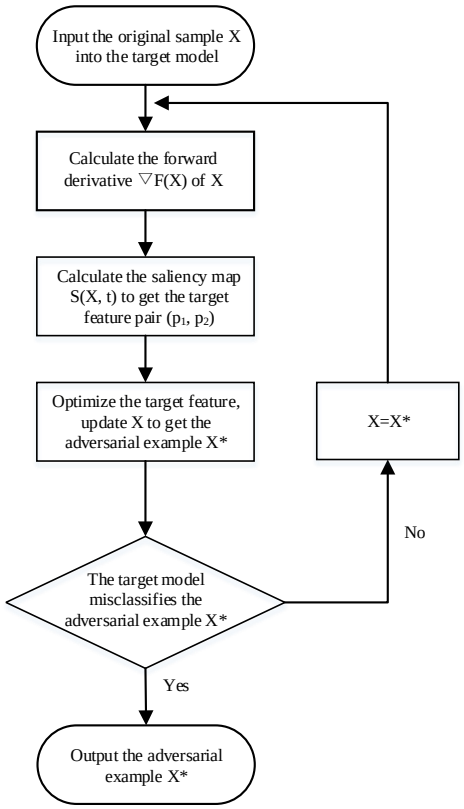


Figure 1: OJSMA flow chart

3.1 Calculating the Forward Derivative of the Target Network

The forward derivative [18] is essentially the Jacobian of the function corresponding to what the target network learned during training. It is the partial derivative of each output class of the last layer of the neural network with respect to the input features. Furthermore, it indicates the degree of influence of each input feature on the target class.

$$\bigtriangledown F(X) = \frac{\partial F(X)}{\partial X} = [\frac{\partial F_j(X)}{\partial X_i}]_{i \in 1..M, j \in 1..N} \tag{5}$$

Equation (5) defines the forward derivative matrix, that is, the derivative of one output neuron F_i with respect to one input dimension X_i.

We start at the first hidden layer of the neural network and can calculate its output in terms of the input components. Then, we recursively calculate each hidden layer in terms of the previous one. Therefore, the forward derivative $\bigtriangledown F$ of a network F can be computed for any input X by successively differentiating layers starting from the input layer until the output layer is reached.

$$\frac{\partial H_k(X)}{\partial x_i} = [\frac{\partial f_{k,p}(W_{k,p} \cdot H_{k-1} + b_{k,p})}{\partial x_i}]_{p \in 1..m_k} \tag{6}$$

where H_k denotes the output vector of the hidden layer k; $f_{k,p}$, the activation function of output neuron p in layer

k; $W_{k,p}$, the weight matrix of the network F; and $b_{k,p}$, the bias of the network F.

3.2 Designing Adversarial Saliency Maps

Saliency maps [21] show the uniqueness of each feature in an input image by simplifying or changing the general image representation to a more analytical style. In adversarial attacks, adversarial saliency map is based on the forward derivatives and is used to show the degree of influence of each input feature on the target class and select the input features with the greatest impact on the target class.

The JSMA algorithm generates two types of adversarial samples using saliency maps with increased image features and reduced image features, respectively. Our OJSMA algorithm proposes an adversarial saliency map that combines both features-increasing and features-decreasing in an input image to select the target features more precisely. Based on the impact of each feature in the input image on the target class, our adversarial saliency map selects some features for which changing their feature values is most beneficial to classify the image as the target class, which may include two types of features, i.e., features for which increasing the feature values is beneficial to the target class and features for which decreasing the feature values is beneficial to the target class. This can be accomplished using the following saliency map $S(X,t)$:

$$S(X,t)[i] = \begin{cases} |\frac{\partial F_t(X)}{\partial X_i}) \cdot \sum_{j \neq t} \frac{\partial F_j(X)}{\partial X_i}|, \\ \qquad \text{if } \frac{\partial F_t(X)}{\partial X_i} \cdot \sum_{j \neq t} \frac{\partial F_j(X)}{\partial X_i} < 0 \\ 0, \text{ otherwise} \end{cases} \quad (7)$$

In Equation (7), so long changing these feature values is beneficial to the image classification as the target class, which includes increasing and decreasing these feature values, the adversarial saliency maps are assigned values $|\frac{\partial F_t(X)}{\partial X_i}) \cdot \sum_{j \neq t} \frac{\partial F_j(X)}{\partial X_i}|$. Then, we select the feature with the maximum value $i_{max} = argmax_i S(X,t)[i]$ in the adversarial saliency maps and decide to increase or decrease the values of its features depending on the sign of $\frac{\partial F_t(X)}{\partial X_i}$.

In the actual experiment, we need to solve two problems. 1) It is difficult to find a single pixel that satisfies the conditions. In layman terms, it is difficult to change an image classification using a pixel. The solution is to search for pairs of pixels at a time since a pixel can compensate for a minor flaw of the other pixel. 2) The success rate $\tau = 64.7\%$ of adversarial example attacks made with reduced image features is relatively low. This is because removing pixels reduces the information entropy, making it difficult for DNNs to extract the information required for sample classification. The solution is to reduce the proportion of pixels with decreasing feature values.

In our experiments, we designed the adversarial saliency map in the following way. In producing the adversarial sample, we counted the pixels with reduced feature values. When the proportion of pixels with reduced feature values is less than s% of the whole pixel, we used the adversarial saliency maps of our algorithm to select target features, as presented in Equation (7). However, when the proportion of pixels with reduced feature values exceeds s% of the total pixels in the image, we used the adversarial saliency maps of Equation (2), that is, we searched only for features whose increased feature values are beneficial for classification into the target class. The pseudocode for the corresponding subroutine saliency map is given in Algorithm 2.

Algorithm 2 Using adversarial saliency maps to find target features

Input:
Forward derivative $\bigtriangledown F(X)$;
Target class t;
The search space τ;
The symbol of α l;
Input image size k;
The ratio to decrease pixel intensities s;
Output:
Target feature pair p_1, p_2

```
Begin
m, max ← 0, l ← 1
for each pixel pair (p, q) ∈ τ do
    α = Σ_{i=p,q} ∂F_t(X)/∂X_i
    β = Σ_{i=p,q} Σ_{j≠t} ∂F_j(X)/∂X_i
    if m < ⌊k·s / (2·100)⌋ then
        if α · β < 0 and −α · β > max then
            (p_1, p_2), max ← (p, q), −α · β
            l = sign(α)
        end if
    else
        if α > 0 and β < 0 and −α · β > max then
            (p_1, p_2), max ← (p, q), −α · β
            l = sign(α)
        end if
    end if
    if l < 0 then
        m++
    end if
end for
End
```

3.3 Optimizing Perturbations and Modifying Samples

The target features were obtained in Section 3.2 using our adversarial saliency maps. This section is the final step of the OJSMA algorithm: modifying the obtained target features, that is, adding perturbations. In the JSMA, the value of the target feature is directly set to the maximum or minimum values, such that overaltered adversarial examples are obtained. Therefore, we defined an objective function to optimize perturbations and obtain adversar-

ial examples with less distortion and greater invisibility. In our experiments, we optimized the objective function using the Adam optimizer [10]. Notably, we need not optimize to accomplish misclassification since we optimize the perturbations to be introduced each time we acquire the desired features, and optimization stops after a certain number of times.

For the differing properties of grayscale and RGB images, two objective functions are suggested. For grayscale images, we optimize perturbations by restricting the L_2 norm and target class's probability:

$$minimize\|m * \delta\|_2^2 + f(X + m * \delta) \tag{8}$$

where δ is adversarial perturbations $\delta = X^* - X$, and m is a constant vector of the same size as the perturbation containing information about the position of the target feature and ensures that only the target feature is optimized. With f defined as:

$$f(X^*) = max\{max_{i \neq t} Z(X^*)_i - Z(X^*)_t, 0\} \tag{9}$$

where X^* is an adversarial example; t, the target class; and Z, the logits of the target network F. The function is used to measure the target class probability. $C(X+\delta) = t$ if and only if $f(X + \delta) \leq 0$, in which case minimizing $f(X+\delta)$ forces the probability of the target class to gradually increase.

For RGB images, we further enhance the invisibility of the perturbation by adding a restriction on the color difference between the original image and adversarial example to the objective function:

$$minimize\|m * \delta\|_2^2 + f(X + m * \delta) + \|LAB(m * \delta)\|_2^2 \tag{10}$$

In many adversarial attack experiments, especially for RGB images, the L_2 norm of perturbations is very small but still visually apparent. Therefore, we further limited the LAB color difference when making adversarial examples of RGB images. The LAB color space is similar to the RGB color space and is also a color model for measuring color, but it is very close to human vision. It is a digital description of human visual perception, and the corresponding color difference calculation is a good indicator of whether two colors are visually similar:

$$\Delta E = \sqrt{(\Delta L)^2 + (\Delta A)^2 + (\Delta B)^2} \tag{11}$$

where L is the pixel brightness and ranges from [0, 100], indicating from black to white. A is the red-green difference and ranges from [127, -128]. B is the blue-yellow difference and ranges from [127, -128]. $\Delta L, \Delta A, \Delta B$ represent the difference between the colors of two pixels in different components, respectively. Generally, the color difference between two pixels is less than 1, $\Delta E \leq 1$ and human vision cannot discern the difference between the colors of two pixels.

To ensure that the modification yields a valid image, we defined a constraint on $\delta : 0 \leq X_i + \delta_i \leq 1$ for all i. We use the box constraint of the C&W attacks [2], and it introduces a new variable ω and instead of optimizing over the variable δ, we apply a change-of-variables and optimize over ω, setting

$$\delta_i = \frac{1}{2}(tanh(\omega_i) + 1) - X_i. \tag{12}$$

Since $-1 \leq tanh(\omega_i) \leq 1$, it follows that $0 \leq X_i + \delta_i \leq 1$; thus, the solution is automatically valid. This approach is seen as a smoothing of clipped gradient descent that eliminates the problem of getting stuck in extreme regions.

4 Experiment

4.1 Experiment Setup

Dataset: In our experiments, we used the MNIST and CIFAR-10 datasets. The MNIST dataset contains 70,000 28×28 greyscale images in 10 classes, divided into 60,000 training images and 10,000 test images. The possible classes are digits from 0 to 9. The CIFAR-10 dataset contains 60,000 $32 \times 32 \times 32$ RGB images. There are 50,000 training images and 10,000 test images. These images are divided into 10 different classes (airplane, automobile, bird, cat, deer, dog, frog, horse, ship, truck), with 6,000 images per class. Figures 2 and 3 present one sample per class, from MNIST and CIFAR-10, respectively.

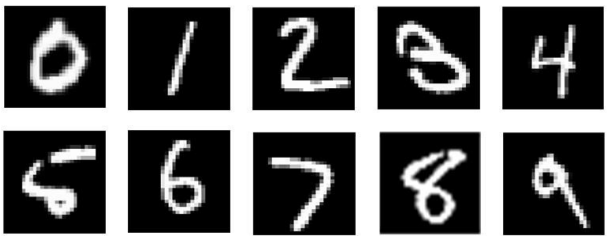


Figure 2: Examples of images from MNIST

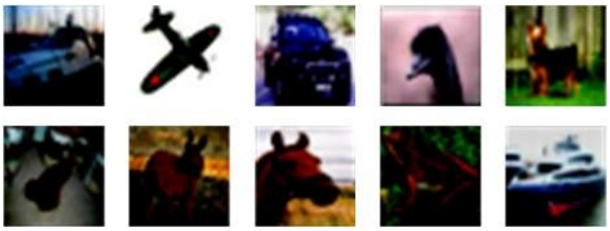


Figure 3: Examples of images from CIFAR-10

Networks: We trained four classical DNNs, i.e., Inception-v3 (Inc-v3) [24], Inception-v4 (Inc-v4) [23], Resnet-v2-152 (Res152) [9], and Inception-Resnet-v2(IncRes-v2) [23], and four ensembles of the before mentioned networks.

Metrics:

1) Accuracy (ACC): This is the percentage of samples correctly classified by the model.

2) Attack success rate (ASR): This is the percentage of successful adversarial examples.

3) Mean L_p distance($L_p - norm$): It is used to measure the size of perturbations [20] and is defined as follows. We chose the mean L_0 norm and the mean L_2 norm to metric perturbation sizes.

$$\|\delta\|_p = (\sum_{i=1}^{n} |\delta_i|^p)^{-p} \tag{13}$$

Where $\delta_i \in (0, 1)$ is the adversarial perturbations of pixel i in the adversarial examples.

4) Transferability: This is the success rate of adversarial examples on an unknown network.

Implementation details: We randomly selected 5000 images correctly classified by the target network from the MNIST and CIFAR-10 test datasets, respectively, as experimental test data. We set a maximum distortion of $\gamma = 14.5\%$ for the MNIST dataset, similar to the setting used in JSMA. We set the ratio of features-decreasing to $s = 3.5\%$ to avoid reducing the number of features too much to reduce the success rate of our attacks. Furthermore, for the CIFAR-10 dataset, we set the maximum distortion to $\gamma = 10\%$, limiting the number of pixels that can be changed in an image to a maximum of 153 pairs, and set the ratio of features-decreasing to $s = 2.5\%$. We searched for pixel pairs at a time and removed modified pixels from the search space. In other words, our algorithm performed a best-first heuristic search without backtracking. Because we reduced the feature size search space with each iteration, this method significantly impacted the performance.

4.2 Training Single Networks and Ensembled Networks

We trained four single networks and four ensembled networks on the MNIST and CIFAR-10 datasets, respectively. Table 1 presents the accuracy of single network on both test datasets. Table 2 presents the accuracy of ensembled network on both test datasets. The sign "-" indicates that the model is removed and the other three models are ensembled, e.g. -Inc-v3 is ensembled of Inc-v4, IncRes-v2, Res153.

On the MNIST dataset, the training epochs were 100 with a batch size of 64. The initial learning rate was set to 0.1 and reduced by a factor of 10 every 20 epochs, and the optimization method was SGD. Similarly, on the CIFAR-10 dataset, the training epochs were 200 with a batch size of 128. The initial learning rate was set to 0.01 and reduced by a factor of 10 every 50 epochs, and the optimization method was SGD.

As can be seen from Table 1 and Table 2, the ensembled network has a higher accuracy rate than the single networks. In Section 4.3, we used the ensembled network as the target network to generate adversarial examples with better transferability.

Table 1: The accuracy of single networks

Dataset	Inc-v3	Inc-v4	IncRes-v2	Res152
MNIST	**99.65**%	99.63%	99.56%	99.62%
CIFAR-10	**94.58**%	93.71%	93.42%	93.96%

Table 2: The accuracy of ensembled networks

Dataset	-Inc-v3	-Inc-v4	-IncRes-v2	-Res152
MNIST	99.71%	99.72%	**99.75**%	99.73%
CIFAR-10	96.28%	96.34%	**96.39**%	96.33%

4.3 Attacking Single Networks and Ensembled Networks

Here, the adversarial examples generated by the JSMA and OJSMA algorithms attacked four single networks and four ensembled networks, and we tested them on all networks. Table 3 presents the attack success rates of the JSMA and OJSMA algorithms on the MNIST dataset, where the diagonal blocks indicate white-box attacks and off-diagonal blocks indicate black-box attacks. In table 4, adversarial examples are generated on an ensemble of three networks, and tested on the ensembled network (white-box setting) and the hold-out network (black-box setting). JSMA+ and JSMA- denote that JSMA generates adversarial samples by increasing image features and decreasing image features, respectively.

Table 3: The success rates on four networks where we attack a single network (MNIST)

Model	Attack	Inc-v3	Inc-v4	IncRes-v2	Res152
Inc-v3	JSMA+	90.3%	12.1%	11.7%	11.3%
	JSMA-	62.3%	6.7%	6.3%	6.1%
	OJSMA	**96.4**%	10.5%	9.6%	9.3%
Inc-v4	JSMA+	17.1%	91.7%	12.2%	11.8%
	JSMA-	7.6%	62.9%	6.5%	6.1%
	OJSMA	14.2%	**96.7**%	10.8%	9.7%
IncRes-v2	JSMA+	17.8%	13.2%	92.3%	12.6%
	JSMA-	7.8%	6.3%	63.2%	6.2%
	OJSMA	15.2%	11.4%	**97.1**%	10.9%
Res152	JSMA+	13.4%	12.9%	12.1%	91.9%
	JSMA-	5.9%	5.7%	6.1%	63.5%
	OJSMA	11.2%	10.8%	10.9%	**97.5**%

Table 4: The success rates of ensemble attack (MNIST)

Model	Attack	-Inc-v3	-Inc-v4	-IncRes-v2	-Res152
Ensemble	JSMA+	87.3%	87.8%	89.7%	88.1%
	JSMA-	56.7%	58.5%	61.2%	60.8%
	OJSMA	**94.5**%	**95.3**%	**96.2**%	**95.4**%
Hold-out	JSMA+	**45.8**%	**46.3**%	**47.5**%	**46.5**%
	JSMA-	29.6%	29.5%	31.8%	30.6%
	OJSMA	42.3%	43.9%	45.5%	44.7%

As can be seen from Table 3, compared with JSMA,

the adversarial examples generated by the OJSMA significantly improved in white-box attack success rates. For example, if adversarial examples are crafted on Inc-v3, the JSMA+, JSMA-, and OJSMA had success rates of 90.3%, 62.3%, and 96.4%, respectively. The transferability of adversarial examples when attacking a single network was relatively poor. However, for an ensembled network, the transferability of adversarial examples from the JSMA and OJSMA had significant improvement. For example, the adversarial example generated on the Inc-v3 attacked the Inc-v4 with success rates of 12.1%, 6.7%, and 10.5% for the JSMA+, JSMA-, and OJSMA, respectively. In contrast, the adversarial example in Table 4 generated on the ensembled network(-Inc-v3) attacked the Inc-v3 with success rates of 45.8%, 29.6%, and 42.3% for the JSMA+, JSMA-, and OJSMA. Thus, compared with the JSMA, our OJSMA algorithm generates adversarial examples with a greater attack success rate, and the usage of the ensembled network significantly enhanced the transferability of the adversarial examples.

Table 5: The success rates on four networks where we attack a single network (CIFAR-10)

Model	Attack	Inc-v3	Inc-v4	IncRes-v2	Res152
Inc-v3	JSMA+	88.7%	8.4%	8.2%	8.0%
	JSMA-	59.6%	3.8%	3.3%	3.1%
	OJSMA	**94.4%**	7.5%	7.3%	6.9%
Inc-v4	JSMA+	12.4%	90.3%	10.2%	9.3%
	JSMA-	5.3%	60.1%	4.6%	4.1%
	OJSMA	10.9%	**94.9%**	8.5%	8.1%
IncRes-v2	JSMA+	13.5%	11.8%	91.2%	10.2%
	JSMA-	6.1%	5.4%	60.9%	5.2%
	OJSMA	12.1%	10.1%	**95.2%**	9.4%
Res152	JSMA+	9.2%	8.8%	8.3%	90.4%
	JSMA-	4.5%	4.2%	3.9%	60.8%
	OJSMA	8.2%	8.0%	7.8%	**95.0%**

Table 6: The success rates of ensemble attack (CIFAR-10)

Model	Attack	-Inc-v3	-Inc-v4	-IncRes-v2	-Res152
Ensemble	JSMA+	85.2%	85.7%	87.6%	86.1%
	JSMA-	54.6%	56.2%	59.1%	58.7%
	OJSMA	**92.3%**	**93.2%**	**94.8%**	**93.9%**
Hold-out	JSMA+	**41.5%**	**43.2%**	**44.1%**	**43.5%**
	JSMA-	26.5%	26.6%	28.1%	27.7%
	OJSMA	40.3%	41.5%	43.3%	43.1%

Table 5 and Table 6 also confirms this conclusion. Table 5 and Table 6 presents the attack success rates of adversarial examples on the CIFAR-10 dataset. The adversarial examples generated by the OJSMA on CIFAR-10 demonstrated a 6% improvement in the success rate of white-box attacks compared with JSMA+. The transferability of adversarial examples from CIFAR-10 datasets has a mean of 30% improvement.

4.4 Measuring Perturbations of Adversarial Examples

In this section, we used the L_0 and L_2 norms to measure the perturbation sizes. Table 7 presents the mean L_0 norm of the adversarial examples for the JSMA and OJSMA on the MNIST dataset, i.e., the average number of modified pixels as a percentage of the total image pixels. Table 8 presents the mean L_2 norm of the adversarial examples for the JSMA and OJSMA algorithms on the MNIST dataset. Table 9 presents the mean L_0 and L_2 norms for both methods on the CIFAR-10 dataset.

Table 7: Mean L_0 norm of adversarial examples (MNIST)

	Inc-v3	Inc-v4	IncRes-v2	Res152	-Inc-v3
JSMA+	4.34%	4.12%	4.08%	4.23%	4.87%
JSMA-	3.42%	3.31%	3.26%	3.38%	3.65%
OJSMA	**4.21%**	**4.03%**	**4.11%**	**4.33%**	**4.46%**

Table 8: Mean L_2 norm of adversarial examples (MNIST)

	Inc-v3	Inc-v4	IncRes-v2	Res152	-Inc-v3
JSMA+	3.41	3.23	3.19	3.31	3.92
JSMA-	2.18	2.27	2.21	2.64	2.87
OJSMA	**1.81**	**1.72**	**1.55**	**1.66**	**2.01**

Table 9: Mean L_0 and L_2 norm of adversarial examples (CIFAR-10)

	Inc-v3		-Inc-v3	
	L_0	L_2	L_0	L_2
JSMA+	5.24%	3.64	5.75%	3.96
JSMA-	3.54%	2.57	3.69%	2.89
OJSMA	**4.23%**	**1.79**	**5.34%**	**1.88**

From these tables, the adversarial examples generated using the two approaches have similar L_0 norms; however, OJSMA has a substantially lower L_2 norm than JSMA+. For example, JSMA+ modifies an average of 4.34% of pixels on the Inc-v3 in Table 6, whereas OJSMA modifies an average of 4.21% of pixels. However, the L_2 norms for JSMA+ are 3.41 and 1.81 for OJSMA, as presented in Table 8. Table 9 shows a similar set of data in the CIFAR-10 dataset. The adversarial examples from JSMA- have relatively low L_0 and L_2 norms. Thus, our OJSMA algorithm provides adversarial examples with minimal L_0 and L_2 norms, demonstrating the usefulness of employing target functions to optimize perturbations in this research.

Figure 4 presents adversarial examples generated by the JSMA+ and OJSMA on the MNIST dataset. Here, the original sample X takes the middle column, adversarial examples generated by JSMA+ on the left, and adversarial examples generated by OJSMA on the right.

The symmetrical position of original samples is adversarial examples of the same target class generated using both algorithms. Visually, the invisibility of the perturbations of the adversarial examples generated by OJSMA is better compared with JSMA+. Figure 5 shows an example of JSMA- generated adversarial examples for different target classes, which have a poorer attack success rate, although the visual perturbation hiding is better.

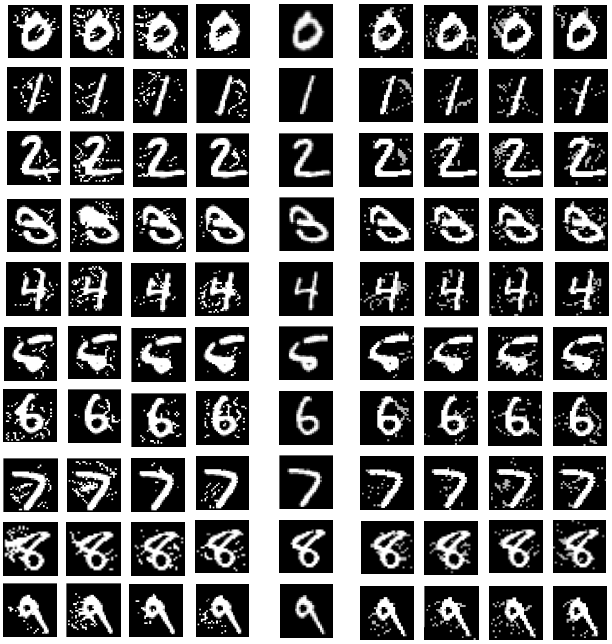


Figure 4: Examples of adversarial examples from JSMA and OJSMA

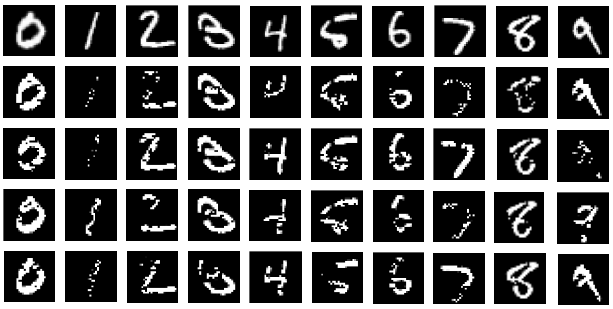


Figure 5: Examples of adversarial examples from JSMA-

Figure 6 presents adversarial examples generated on the CIFAR10 dataset. The first, second, and third rows show the original samples, adversarial examples generated by JSMA+, and adversarial examples generated by OJSMA, respectively. Compared with JSMA, the adversarial samples generated by OJSMA are visually more similar to the original samples. This is because we add the LAB color difference to the objective function, which further reduces the L_2 norm of the perturbation based on the L_0 norm.

Figure 6: Examples of adversarial examples from JSMA and OJSMA

4.5 Comparing OJSMA and Other Targeted L_0 Attacks

In this section, we focus on other targeted L_0 attacks, including other variants of JSMA and C&W L_0 attacks [2]. We also focus on MJSMA [28] and One Pixel attack [22], but both approaches have proven to be very effective for non-targeted uses, so we do not compare them here. Table 10 compares OJSMA with JSMA+, WJSMA [4], TJSMA [4], and C&W L_0 attack objectively in four dimensions: attack success rate (ASR), L_0 norm (L_0), L_2 norm(L_0), and time overhead (T). Table 11 shows the running time comparison of these five L_0 attack algorithms to generate one adversarial example on MNIST. The target model we use in this comparison experiment is Inception-v3. We preferred to work with the DNN model as this makes the paper shorter and moreover, it values more our approach giving us more advantage with respect to JSMA.

Table 10: The comparison experiment of OJSMA and other L_0 attacks.

	MNIST			**CIFAR-10**		
	ASR	L_0	L_2	ASR	L_0	L_2
JSMA+	90.3%	4.34%	3.41	88.7%	5.24%	3.64
OJSMA	96.4%	4.21%	**1.81**	94.4%	4.23%	**1.79**
WJSMA	95.8%	4.32%	3.32	94.1%	4.72%	3.42
TJSMA	96.3%	4.25%	3.24	94.8%	4.62%	3.24
C&W	**99.2**%	**3.18**%	2.83	**99.1**%	**3.85**%	3.02

Table 11: Run-time comparison(MNIST)

	JSMA	**OJSMA**	**WJSMA**	**TJSMA**	**C&W**
T(s)	4.12	38.9	4.01	**3.84**	385.8

Our method is comparable to other excellent variants (WJSMA, TJSMA) in terms of attack success rate, but significantly smaller adversarial perturbations. This is because our method optimizes the L_2 norm of perturbations of selected features, which also causes it to lose more in time overhead compared to WJSMA and TJSMA. Our method and all JSMA and variants are inferior in terms of success rate compared to C&W L_0 attacks, but the C&W L_0 attack time overhead is very high. In summary, our algorithm sacrifices some time but is faster than the

C&W L_0 attack, has a significant improvement in success rate and perturbation compared to the original JSMA, and has an advantage in perturbation compared to other JSMA variants.

5 Conclusion

In this paper, we introduced Optimized Jacobian-based saliency map attack (OJSMA), a new optimized JSMA for targeted and white-box attacks on image classification models. OJSMA generates more robust adversarial samples by introducing new saliency maps, optimizing perturbations, and adding an ensembled network, starting from three aspects, namely, the attack success rate, invisibility of perturbations, and transferability. Experimental results indicate an average improvement of 6% in the success rate of OJSMA compared with JSMA. Furthermore, our attacks considerably reduce the L_0 norm of perturbations while maintaining a tiny L_2 norm, which enhanced the invisibility of perturbations visually. We compared OJSMA with other targeted L_0 attacks, including other JSMA variants (WJSMA, TJSMA) and C&W L_0 attacks. Our algorithm sacrifices some time but is faster than the C&W L_0 attack, has a significant improvement in success rate and perturbation compared to the original JSMA, and has an advantage in perturbation compared to other JSMA variants. In addition, we used the ensembled network as the target network for JSMA and OJSMA, and both algorithms showed an average improvement of 30% in the transferability, demonstrating the effectiveness of the ensembled network in improving the transferability of the adversarial examples.

we should mention that despite improving JSMA, our approach is still not scalable to large datasets. This is because of the high computational cost of saliency maps when the dimension of inputs becomes large. Our approach is therefore intended for "small" datasets such as those considered in the paper. Nevertheless, this kind of datasets is very common in real-life applications. See also the recent paper [8]. This paper finds that the transferability of adversarial examples is low, and its improvement by our method is limited. Therefore, in future research, we will continue to investigate the transferability of adversarial examples to improve their generalization between models.

Acknowledgments

This work was supported by the Natural Science Foundation of China under Grant 61562065 and the Inner Mongolia Natural Science Foundation Project under Grant 2019MS06001.

References

[1] A. Athalye, L. Engstrom, and A. Ilyas, "Synthesizing robust adversarial examples," in *The 35th International Conference on Machine Learning (ICML'18)*, pp. 284–293, Sweden, 2018.

[2] N. Carlini and D. Wagner, "Towards evaluating the robustness of neural networks," in *2017 IEEE Symposium on Security and Privacy (SP'17)*, pp. 39–57, USA, 2017.

[3] P. Chen, H. Zhang, Y. Sharma, J. Yi, and C. Hsieh, "Zoo: Zeroth order optimization based black-box attacks to deep neural networks without training substitute models," in *AISec'17: Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security*, pp. 15–26, Texas, USA, 2017.

[4] T. Combey, A. Loison, and M. Faucher, "Probabilistic Jacobian-Based Saliency Maps Attacks," in *Machine Learning and Knowledge Extraction*, vol. 2, pp. 558–578, 2020.

[5] Y. Ding, Z. Wang, B. Li, G. Xu, and L. Deng, "Automatic small target detection in complex background: a state of-the-art survey," in *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, vol. 11884, p. 118841S, 2021.

[6] Y. Dong, F. Liao, T. Pang, H. Su, J. Zhu, X. Hu, and J. Li, "Boosting adversarial attacks with momentum," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9185–9193, Salt Lake City, UT, USA, 2018.

[7] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," *Statistics*, vol. 3, 2014.

[8] H. Hajri, M. Césaire, T. Combey, S. Lamprier, and P. Gallinari, "Stochastic sparse adversarial attacks," in *2021 IEEE 33rd International Conference on Tools with Artificial Intelligence (ICTAI'20)*, Washington, DC, USA, 2020.

[9] K. He, X. Zhang, S. Ren, and S. Jian, "Identity mappings in deep residual networks," in *Computer Vision (ECCV'16)*, vol. 9908, pp. 630–645, 2016.

[10] D. Kingma and J. Ba, "ADAM: A method for stochastic optimization," in *Computer Science*, 2014.

[11] A. Kurakin, I. Goodfellow, and S. Bengio, "Adversarial examples in the physical world," in *Statistics*, vol. 2, 2016.

[12] Y. Liu, X Chen, C. Liu, and D. Song, "Delving into transferable adversarial examples and black-box attacks," in *The 5th International Conference on Learning Representations (ICLR'17)*, Toulon, France, 2017.

[13] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," in *The 6th International Conference on Learning Representations (ICLR'18)*, Vancouver, BC, Canada, 2018.

[14] A. Modas, S. M. Moosavi-Dezfooli, and P. Frossard, "Sparsefool: A few pixels make a big difference," in *2019 IEEE/CVF Conference on Computer Vision*

and Pattern Recognition (CVPR'19), pp. 9079–9088, Long Beach, CA, United states, 2019.

[15] S. M. Moosavi-Dezfooli, A. Fawzi, and P. Frossard, "Deepfool: a simple and accurate method to fool deep neural networks," in *The 29th IEEE Conference on Computer Vision and Pattern Recognition (CVPR'16)*, pp. 2574–2582, Las Vegas, NV, United states, 2016.

[16] N. Narodytska and S. Kasiviswanathan, "Simple black-box adversarial perturbations for deep networks," in *IEEE Computer Vision & Pattern Recognition Workshops*, pp. 1310–1318, Honolulu, HI, USA, 2017.

[17] M. Ohta, N. Berger, A. Sokolov, and S. Riezler, "Sparse perturbations for improved convergence in stochastic zeroth-order optimization," in *Machine Learning, Optimization, and Data Science*, pp. 39–64, Cham, 2020.

[18] N. Papernot, P. McDaniel, S. Jha, M. Fredrikson, Z. B Celik, and A. Swami, "The limitations of deep learning in adversarial settings," in *2016 IEEE European Symposium on Security and Privacy (EuroS P'19)*, pp. 372–387, Saarbruecken, Germany, 2016.

[19] G. Shao, L. Tang, and H. Zhang, "Introducing image classification efficacies," in *IEEE Access*, vol. 9, pp. 134809–134816, 2021.

[20] Y. Shi, S. Wang, and Y. Han, "Curls & whey: Boosting black-box adversarial attacks," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR'19)*, pp. 6512–6520, Long Beach, CA, USA, 2019.

[21] K. Simonyan, A. Vedaldi, and A. Zisserman, "Deep inside convolutional networks: Visualising image classification models and saliency maps," in *The 2nd International Conference on Learning Representations (ICLR'14)*, Banff, AB, Canada, 2014.

[22] J. Su, D. V. Vargas, and K. Sakurai, "One pixel attack for fooling deep neural networks," in *IEEE Transactions on Evolutionary Computation*, vol. 23, pp. 828–841, 2019.

[23] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning," in *31st AAAI Conference on Artificial Intelligence (AAAI'17)*, p. 4278–4284, San Francisco, California, USA, 2017.

[24] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR'16)*, pp. 2818–2826, Las Vegas, NV, USA, 2016.

[25] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus, "Intriguing properties of neural networks," in *The 2nd International Conference on Learning Representations (ICLR'14)*, Banff, AB, Canada, 2014.

[26] K. Wang, L. Lozano, D. Bergman, and C. Cardonha, "A two-stage exact algorithm for optimization of neural network ensemble," pp. 106–114, 2021.

[27] X. Wang, A. Bao, Y. Cheng, and Q. Yu, "Multipath ensemble convolutional neural network," in *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 5, pp. 298–306, 2021.

[28] R. Wiyatno and A. Xu, "Maximal jacobian-based saliency map attack," in *Machine Learning*, 2018.

[29] C. Xie, J. Wang, Z. Zhang, Z. Ren, and A. Yuille, "Mitigating adversarial effects through randomization," in *The 6th International Conference on Learning Representations (ICLR 2018)*, vol. Vancouver, BC, Canada, 2018.

[30] C. Xie, Z. Zhang, Y. Zhou, S. Bai, and A. L. Yuille, "Improving transferability of adversarial examples with input diversity," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR'19)*, pp. 2725–2734, Long Beach, CA, USA, 2019.

Biography

Wenwen Zhang was born in Heze, China, in November 1996. She received the B.S. degree in computer science and technology from the Jining Medical University, in 2019. She is currently pursuing the master's degree in computer science and technology with the Inner Mongolia University of Science and Technology. Her research interests include adversarial attacks based on image classification.

Xiaolin Zhang was born in Baotou, China, in December 1966. She received the bachelor's degree in computer science and technology from Northeastern University, in 1988, the master's degree in automation from the Beijing University of Science and Technology, in 1995, and the Ph.D. degree in computer science and technology from Northeastern University, in 2006. Since 1988, she has been with the Inner Mongolia University of Science and Technology, where she is currently the Deputy Director of the Head of the Computer Science Department, Professor Committee of the Information Technology College, and the Director of the Department of the Computer Science. She has trained more than 70 master's degree students and now is training 12 master's degree students. She has published over 80 academic articles, including more than 30 articles in EI and 7 articles in SCI. She is responsible for many projects, such as the National Natural Science Foundation of China, the National Social Science Fund Project, the Chunhui Project of the Ministry of Education, the Natural Science Foundation of Inner Mongolia Project, and the Inner Mongolia Education Department Fund Project. Her current research interests include image processing, natural language processing, adversarial attacks of image and text, machine learning security, big data processing technology, social network privacy protection technology. Dr. Zhang is a member of the Chinese Computer Society, the Information System Professional Committee, the China Computer Society, and the Director of the Inner Mongolia Autonomous

Region Computer Society.

Kun Hao is currently engaged in post-doctoral research in the College of Medicine and Biological Information and Engineering, Northeastern University, China. He obtains the Ph.D. degree with the School of Computer Science and Engineering, Northeastern University, China, 2021. His research interests include machine learning security, blockchain database, big data management, and medical informatics.

Jingyu Wang was born in Kaifeng, Henan Province in 1976. He graduated from the Department of Metallurgical Engineering of Inner Mongolia University of Science and Technology with a bachelor's degree in 1999, from the Department of Computer Science of College of Information Engineering of Inner Mongolia University of Science and Technology with a master's degree in 2002, and from the Department of Computer Science of College of Computer and Communication Engineering of University of Science and Technology Beijing with a doctorate degree in 2014. He is currently the Dean of School of Information Engineering of Inner Mongolia University of Science and Technology, undertaking scientific research projects of National Natural Science Foundation of China, Inner Mongolia Natural Science Foundation, Inner Mongolia Education Department, etc. Currently, he has published more than 30 academic papers in relevant journals at home and abroad. His main research interests are big data and security, blockchain and security.

Shuai Zhang was born in Dongying, China, in November 1996. He received the B.S. degree in computer science and technology from the Inner Mongolia University, in 2019. He is currently pursuing the master's degree in computer science and technology with the Inner Mongolia University of Science and Technology. His research interests include adversarial attacks and defense based on image classification.

A Software-Defined Security Framework for Power IoT Cloud-Edge Environment

Rixuan Qiu[1], Yu Fu[1], Jian Le[2], Fuyong Zheng[1], Gan Qi[2], Chao Peng[1], and Yuancheng Li[3]
(Corresponding author: Jian Le)

State Grid Jiangxi Information & Telecommunication Company, Nanchang, China[1]
School of Electrical Engineering and Automation, Wuhan University, China[2]
Wuhan 430072, China
School of Control and Computer Engineering, North China Electric Power University, Beijing, China[3]
Email: ncepua@163.com
(Received Apr. 20, 2022; Revised and Accepted Oct. 12, 2022; First Online Oct. 15, 2022)

Abstract

The application of cloud computing brings many security challenges to the power Internet of things, such as DoS attacks, location-based attacks, man-in-the-middle attacks, and sniffing. Aiming at the cloud-edge security of power Internet of things, we propose SD-PIoT, a framework based on the software-defined perimeter (SDP) and software-defined network (SDN). In the framework, SDP and SDN jointly enhance the security of the cloud edge in a complementary way. SDP protects the security of the cloud and the inner layer of SDN by rejecting all unauthorized edge traffic. SDN deploys SDP applications to the SDN application plane to realize SDP's flexible deployment and management. In addition, SDN improves the reliability of network communication by controlling the topology. The security, communication reliability, and performance are analyzed through simulation experiments. The results show that the scheme can effectively resist network attacks, improve communication reliability and performance, and improve the cloud-edge protection ability of power Internet of things.

Keywords: Cloud-Edge Environment; Power IoT; Software-defined Network; Software-defined Perimeter

1 Introduction

With the advent of the information age, tens of billions of power equipment are connected to the Internet, and a large amount of data on the network is transmitted between terminal devices [19]. In 2006, the cloud was proposed as a remote server for power companies, providing additional storage and data processing services. After the birth of cloud computing, paradigms such as edge computing and fog computing emerged, acting as the intermediary between edge devices and the cloud to improve the performance of the IoT. Gartner predicts that by 2025, more than 90% of enterprises will adopt cloud infrastructure and platform strategy [8], which greatly reduces the burden on power companies. Although cloud computing and other paradigms derived from it provide many advantages to the power IoT, some security challenges will inevitably arise in the cloud-edge environment. Reference [12]concludes by analyzing different types of cloud that although cloud computing itself has advantages, it also faces the danger of information leakage brought by insecure visitors. Reference [9]summarizes the security issues of cloud computing into four types of threats, including data level, privacy level, user level and provider level. It indicates that these threats come from attacks. Reference [10] analyzes the insecurity of traditional network boundaries to cloud computing servers and uses SDN technology to manage and control the network to alleviate cloud security problems. Reference [11] found that, due to many edge devices in the power IoT, the probability of attackers discovering terminal devices with defective network protocols increases, which will lead to these defective devices being used by hackers to launch DoS attacks on the cloud.

On the one hand, internal enterprise data is stored in the enterprise cloud, and the vast majority of enterprise employees need to have access and processing rights to these data. This expansion of access scope increases the probability of malicious "internal employees". On the other hand, edge/fog computing leverages several different techniques to build networks, introducing the possibility of multiple attacks, such as man-in-the-middle attacks, wireless jamming, and DoS attacks [18]. For the former, malicious "internal employees" and man-in-the-middle attacks threaten the confidentiality and integrity of information, and wireless jamming consumes bandwidth, spectrum, and computing resources at the edge. But in contrast, DoS attacks can occur in many ways, such as flooding, redirection, jamming, and spoofing, which are simple and effective. They can quickly lead to network confiden-

tiality, integrity, and availability loss.

The power IoT cloud-edge environment requires a new framework to address the above security issues. Applying SDP technology in networks requiring remote interaction is a new and active area of research. References [14, 15, 19–21] all demonstrate the security advantages of SDP technology applied in environments such as the Internet of Things, which can resist various attacks, including DoS, and alleviate the security challenges caused by traditional boundary blurring. However, these studies did not address the flexible deployment and scalability of SDP applications, such as how to react when new devices are added to the network, nor did they consider the enterprise's network configuration. In addition, the literature [14] discussed that the challenges faced by SDP had not been solved, such as network interruption and configuration update. SDN separates the data plane and the control plane in a software-defined way, providing new ideas. SDN controls a global view of the network at the control plane in a software-defined way. The programmability of the SDN application layer brings more possibilities to improve the network in the power IoT.

Therefore, this paper proposes a software-defined power Internet of Things security protection framework SD-PIoT, which aims to help the power Internet of Things defend against cloud-side network attacks, simplify network management, and additionally improve the reliability of cloud-side communication under SDN's advantages. The main contributions of this paper are summarized as follows:

1) Aiming at the above security issues, we propose an SD-PIoT security framework, which uses SDP to protect the cloud-side security and introduces SDN to manage the network to alleviate the limitations of SDP.

2) The framework integrates the two by placing the SDP controller at the SDN application plane to provide the required security protection technology.

3) The solution has been tested for attack and network performance. The results show that the framework can protect the cloud-edge environment from attacks, improve communication reliability, and negligible network delay.

The rest of the paper is structured as follows. In the second section, some related works mentioned in the article are expounded. In Section 3, We described the SDP and SDN architectures. And a security framework SD-PIoT based on SDP and SDN is proposed and described. In Section 4, test implementation and result evaluation are presented. Finally, we summarize the proposed solution in Section 5.

2 Related work

Even applications in leading cloud service providers are not guaranteed to be 100% free from attacks [7, 16] . Due to a large number of terminal devices at the edge of the power IoT, the security of the cloud faces greater threats, such as privacy theft, resource fraud, and DoS attacks [13, 26]. Traditional network tools are inefficient for the massive power data collection, storage, processing, and forwarding necessary in power IoT networks. Inherent uncertainties such as packet loss and communication outages hidden at the edge of the network greatly degrade the quality of service (QoS). Therefore, the power Internet of Things needs a more advantageous security framework to improve the security of the cloud edge environment.

SDP has been explored in IoT security protection as a relatively novel security architecture in recent years. Reference [15] applies SDP to IoT applications in Message Queuing Telemetry Transport (MQTT) to provide an additional layer of security by exploiting the network stealth properties of SDP. In Ref. [19], SDP addresses the security challenges of edge computing in IoT, insuring the cloud from edge traffic. Reference [21] applies the logical boundaries of SDP to narrow the scope of network access and connectivity of network virtual functions (VNFs) to trusted identities. Reference [20] discusses the security issues cloud infrastructure-as-a-service faces and proposes an SDP-based solution that allows only authorized clients to access protected services. The existing literature has used the SDP controller's authentication, dynamic authorization and other programs to achieve corresponding security protection. But they have not specifically discussed the deployment and management of these programs in the network and have not considered the program expansion that enterprises need.

A lot of work has been done to study the network management advantages of SDN. Reference [2] proposed a new IoT security architecture based on SDN, which helps to improve the security and communication reliability of IoT. Reference [22] developed a powerful control and network management platform using SDN in the smart grid scenario. Reference [5] used the centralized control logic of SDN to alleviate the edge uncertainty of the Internet of Vehicles and used the edge offload delay as an indicator to verify the scheme's effectiveness. Rohit *et al.* [4] introduced SDN into IoT to improve resource-constrained IoT low-power devices' network performance and evaluated the scheme according to latency and packet loss rate. SD-MIoT [23] is an SDN solution applied to the mobile IoT environment, which uses SDN to alleviate the communication problems caused by mobile nodes, and uses the message passing success rate to measure the reliability of the scheme.

However, there are still some security problems to be solved in SDN. For example, the open programmability of SDN brings a trust crisis. The global control of the SDN controller makes the network more vulnerable to denial of service (DoS) attacks. In addition, the internal SDN faces the security risk of the controller being invaded. The separation of control plane and data plane brings security challenges to the outer and inner layers of Sdn. The outer security challenge means that the server or switch is vul-

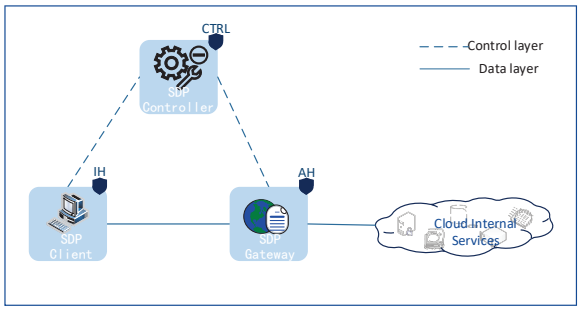


Figure 1: SDP client-gateway architecture

nerable to flood attacks, which may lead to the collapse of the entire security system. Inner security challenge refers to that intruders attack SDN controllers, generate false network data, and launch different attacks on the whole network. Reference [17] introduces the SDP framework to improve the outer layer security of SDN-based networks and discusses the interesting integration points between the two. It demonstrates the integrability of SDP and SDN, which lays the foundation for this paper, but the security issues within SDN are not addressed.

The difference between this paper and previous research is that SDP and SDN are cleverly combined to complement each other's shortcomings, bringing huge security advantages to the cloud-edge environment of the power Internet of things.

3 Method

3.1 Software-Defined Perimeter (SDP)

The Cloud Security Alliance (CSA) published the first SDP specification in 2014 [1], and SDP has attracted much attention in academia. SDP follows the idea of zero trust "continuous verification, never trust", only authenticated identities are allowed to access services, and unauthenticated identities remain in a state of "network stealth" [3]. As shown in Figure 1, the SDP architecture consists of three main components, the SDP controller, the SDP initiating host (IH), and the SDP accepting host (AH). The SDP controller is located in the control plane and acts as the SDP brain, which determines any host's grant and access rights; the SDP IH is the host that initiates the service request; the SDP AH is the host that accepts the service request connection. After the IH is authenticated at the controller, it can connect and access the AH. Therefore, the SDP structure can effectively mitigate many network attacks that traditional boundary-based isolation protection methods cannot be solved, including server scanning, DoS and man-in-the-middle attacks [14].

Various deployment models for SDP include client-gateway, client-server, client-server-client, etc [1]. For the problem to be solved, this paper chooses the client-gateway model. In this model, the edge device acts as the client IH, the gateway acts as the AH, and the services and resources in the cloud are hidden behind the gateway to allow only connections from the gateway, which can make the cloud achieve "network stealth" effect.

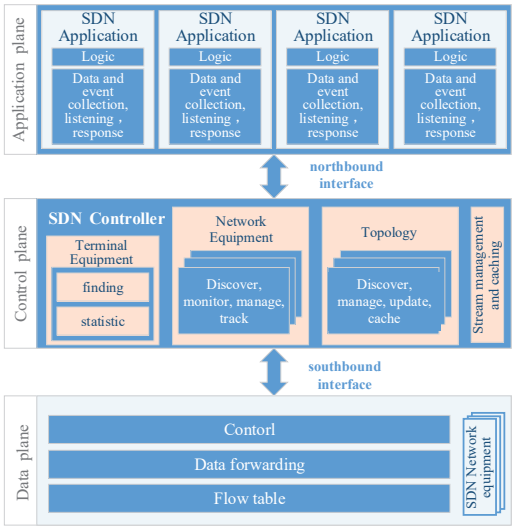


Figure 2: SDN architecture

3.2 Software-Defined Network (SDN)

With the development of network virtualization, SDN is proposed as an implementation of network virtualization [25]. As shown in Figure 2, the hierarchical structure of SDN consists of three planes, namely the application plane, control plane and data plane. The main components of these three planes are SDN application, SDN controller, and SDN network device, respectively.

In the control plane, the SDN controller has logical centralization and programmability. And it maintains the overall information of the global network, which is convenient for operators and researchers to implement different policy decisions through upper-layer programming. On the data plane, an SDN network device with a data forwarding function stores a flow table in which different forwarding rules are stored for different types of data packets. After receiving a data packet, the SDN device will search the entire flow table in descending order of priority and process the data packet according to the matching flow table entry rules [25]. The application plane contains various SDN applications, which run on the SDN controller. Events from the lower layers will trigger their different responses to the SDN controller, allowing operators to program and deploy new applications without caring about the details of the lower layers. Therefore, SDN can significantly simplify network control and enable flexible and efficient management of SDP-introduced power IoT networks.

3.3 Proposed Software-Defined Framework

This section will introduce the flexible combination and complementation of SDP and SDN, and the protection and control capabilities they provide for the cloud-edge environment of the power IoT. Figure 3 depicts the overall design of our proposed framework SD-PIoT, which extends the SDN architecture. For the cloud-side security of the power IoT, we add an edge device plane and a cloud center plane to the original SDN architecture.

3.3.1 Framework description

The framework contains five main components: SDP client (IH), SDP controller (CTRL), SDP gateway (AH), SDN controller and SDN switch.

1) SDP client (IH): The edge device in the network that needs to access the cloud is used as the IH in the SDP, and the SDP client software is installed on it. After the edge device goes online, it can communicate with other components.

2) SDP controller (CTRL): In this solution, the SDP controller runs on the SDN controller as an SDN application. It monitors the SDN controller's status, responds to the events, and determines whether every IH can access the AH. For clarity, the SDP controller is denoted by CTRL after this.

3) SDP gateway (AH): The gateway, as an AH, obeys the arrangement of the SDP controller, initially configures the firewall rule as "deny all", provides access connections to cloud services for authorized identities, and monitors and records the entire connection process.

4) SDN controller: The SDN controller holds a global view of the entire network, controls the underlying network consisting of switches, and provides an abstraction of the underlying network resources to SDN applications such as CTRL to wake them up and make them responsive.

5) SDN switch: A SDN device at the network data plane. The SDN switch forwards the data (in packet units) of other components. A flow table is stored in the switch, and each item of the flow table consists of matching fields and actions. The switch decides the removal and retention of incoming packets according to the flow table and the instructions of the SDN controller.

Application Plane: There are SDN applications (such as Graphical User Interface, routing procedures) required by enterprises on the application plane. Different SDN applications implement corresponding network functions. SDN applications can execute the assigned functions by manipulating the SDN controller when an event from the control layer or an external input drive is detected.

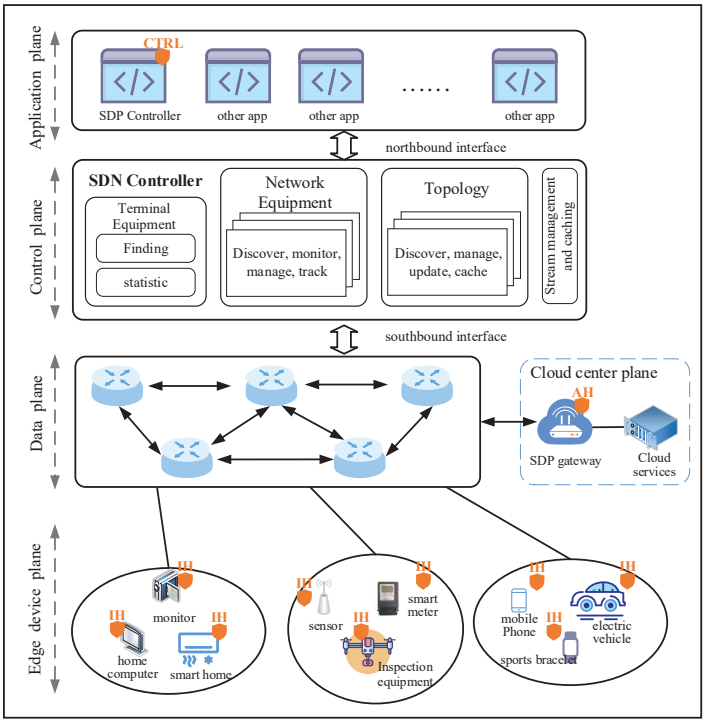


Figure 3: SD-PIoT model

Users/developers can manage and control the network without touching the underlying network by programming different SDN applications. In addition to basic SDN applications, power companies can design different SDN applications according to their needs to realize network programmability. It is worth mentioning that CTRL can be a program [6], so this paper places CTRL as an SDN application in the application layer. After receiving the access request from the IH, the CTRL verifies whether the device completes the SPA authorization. If it passes the verification, it instructs the SDN controller to forward the request to the AH. Otherwise, instructs the SDN controller to discard the packet. The authentication of CTRL is shown in Algorithm 1.

Control plane: The control plane has a logically centralized SDN controller that maintains a global view of the entire network, implements various policy decisions, and facilitates efficient and reliable communication services. In addition to the SDN controller, the control plane also includes northbound interfaces and southbound interfaces, which are respectively open to the application plane and the data plane. As the brain of SDN, the SDN controller realizes centralized logic control, provides the application plane with an abstract model of the underlying network, including states and events through the northbound interface, and wakes up SDN applications to perform corresponding functions. In addition to the above, the control plane embodies the application plane's request to configure, manage, and control the data plane according to its logic.

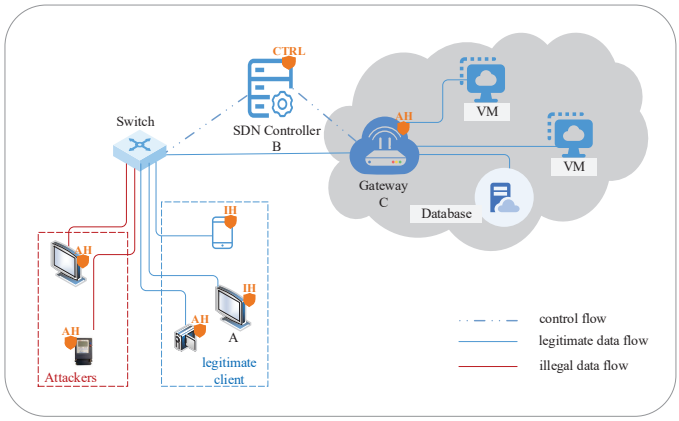


Figure 4: Case of the scheme

Data plane: As shown in Figure 3, this plane consists of SDN network devices (switches are used in this paper) with data forwarding and processing functions. All components in the framework are connected through switches. All switches in the network work under the control of the SDN controller, and the data layer communicates with the control layer through the southbound interface of the control layer. A forwarding table resides in each switch. When the switch receives a packet, it will search for the forwarding table in priority order. If it finds a flow entry that matches the packet, it will execute the configured action. If no match is found, the packet is discarded or forwarded to the controller (the latter is used in this paper).

Edge device plane: The network edge devices of the Power IoT all reside in this layer, such as sensors, detectors, smart meters, mobile devices, etc. Edge devices in this layer are mostly heterogeneous, come from different networks and use different communication technologies such as cellular, WiFi, and RFID. These devices are configured with IH and have the function of applying for cloud access. When edge devices want to access cloud services, they need to send SPA packets to CTRL for authentication. An unauthenticated IH will not be granted access. Algorithm 1 is an edge device authentication.

Cloud center plane: Many services and resources need to be protected in the cloud. These service resources are hidden behind the gateway. For IHs without CTRL authentication, the services in the cloud are inaccessible. Therefore, the cloud has remained "Network stealth" to potential attackers.

3.3.2 Scheme Process

Figure 4 shows the specific use case of the scheme.

1) Initialization: A default flow f0 with priority 0 is set on all switches. The task of flow f0 is to forward all traffic from the edge to the SDN controller. Traffic is forwarded between switches in two paths, optimal and redundant, as determined by the SDN controller.

Algorithm 1 Edge Device Authentication

The network has been deployed;
The basic format of the flow entry is $(priority, source, action, duration)$;
Add default flow $f_0(0, all-edge-devices, forwarded-to-SDN-controller, all-time)$ to SDN switch;
AH is online and authorized by CTRL;
IH (device id is u_0) goes online and sends an authentication request Req to CTRL;

if Req is a valid SPA package **then**
  CTRL verifies u_0 certificate and key;
  if certificate and key are verified **then**
    CTRL determines the accessible cloud service list l_0 of u_0;
    CTRL instructs the SDN switch to add flow $f_1(10, u_0, forwarded-to-AH, k)$ and the k value is adjusted by the power company according to the situation;
    CTRL makes l_0 as a message and sends it to the IH, informing it of the list of cloud services it can access;
    CTRL sends l_0 to AH, informing it to update firewall rules and allow IH to access services in the list;
    AH updates firewall rules after receiving l_0, and allows IH to communicate with services in l_0 within time k;
  else
    CTRL commands the SDN controller to drop Req;
    CTRL instructs the SDN switch to add a flow $f_2(11, u_0, drop, k/2)$ to punish u_0 for not being able to authenticate within $k/2$ time;
  end if
else
  CTRL commands the SDN controller to drop Req;
  CTRL instructs the SDN switch to add a flow $f_2(11, u_0, drop, k/2)$ to punish u_0 for not being able to authenticate within $k/2$ time;
end if

The firewall configuration of the SDP gateway is initialized to "deny all".

2) Authentication: The newly online IH(A) sends an authentication request to CTRL. Since the switch sets the default forwarding flow f0, the request packet finally reaches the SDN controller (B). CTRL on B detects the authentication request packet, checks A's credentials and verifies its identity and determines its access level:

   a. If the verification is passed, CTRL instructs B to send a message to the SDP gateway AH (C), and instructs C to add a new firewall rule h1 to

allow communication between A and C. At the same time, a new flow entry f1 with a priority greater than 0 is added to the switch. Within the specified time T, the service request packet from A is forwarded to C by default. After T time, the flow entry f1 and the new firewall rule h1 are automatically deleted.

b. If verification fails, CTRL instructs the SDN controller to discard the packet.

3) Request for service: Authorized A requests C to access services in the cloud. The switch forwards the request to C according to the priority without going through B, and C checks whether the request is a SPA packet:

a. If the data packet is correct, C allows A to access some cloud services according to its own firewall rule h1.

b. If the data packet is incorrect, C reports the relevant information to B and deletes the firewall rule h1 related to A. After the CTRL on B detects the message, it immediately instructs the switch to delete the related flow entry f1, and A will need to re-authenticate the CTRL.

During the A and C connection process, C monitors and records A's behavior through the tracking mechanism.

3.3.3 Network Management and Control

In the cloud-edge interaction process of the power IoT network, the SDN controller is located in the control plane to monitor the global network and regularly collect information to update the network. The SDN controller can discover the newly added edge terminal equipment and switch network equipment for the first time. After discovering a new edge device, CTRL issues a certificate and user key for the device, and the SDN controller records the device's communication protocol. After discovering network devices such as switches, the SDN controller updates network topology, sets default flows, and maintains and tracks network status information. When the traffic from the new edge device enters the network, the controller determines the transmission technology and routing path according to the statistical information and adds a flow table entry suitable for the device in the switch to facilitate subsequent traffic forwarding. The control plane monitors the network in real-time and updates it regularly, which greatly improves the scalability of the network.

In order to prevent communication interruption, packet loss, and ensure real-time communication between the cloud and the edge, this solution takes advantage of the SDN controller to monitor the network topology in real-time, and forwards legitimate traffic through the optimal path and another redundant path to improve communication reliability. At the same time, SD-PIoT opens up the application plane's programmability to power enterprises. Enterprises can create their new applications according to their internal needs without caring about the differences between the underlying devices. Applications can modify the underlying forwarding rules through the northbound interface to achieve rapid network configuration and deployment, simplifying network management. For example, in zero trust, security protection includes several basic processes: user identity management, dynamic authorization control, and access control [24]. Suppose the device fails to authenticate many times during the service request. In that case, the enterprise can program an application to evaluate the trust degree of such identities and then authorize its access level according to the trust degree. The upper-layer pluggable application of the SDN controller can be programmed according to the internal requirements of the enterprise and the records of CTRL and AH in the authentication and service request phases, which realizes the application's scalability.

In addition, all edge traffic reaching the SDN controller in the network will first be verified by CTRL to decide whether to retain the traffic. If the verification fails, the corresponding packet will be directly discarded. This effectively prevents intruders from damaging the SDN controller and ensures the inner layer security of the SDN.

The proposed framework, SD-IoT, protects cloud-side security and achieves the flexibility, scalability, and programmability of the network. It ensures reliable communication between heterogeneous devices and the inner layer security of SDN, and simplifies network management.

4 Experiment and Analysis

We evaluate the proposed framework from security, communication, reliability, and performance aspects. The main goal of this framework is to improve cloud-based IoT security and communication reliability without compromising network performance.

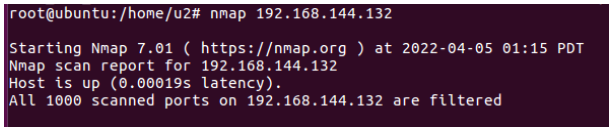
4.1 Testbed Environment

The testbed consists of SDP and SDN. The SDP part uses Waverley Labs' Open SDP project, and the SDN part uses public OpenFlow components. The simulation experiment uses six Linux Ubuntu16.04 virtual machines representing two edge devices, an SDN controller running the CTRL application, an SDP gateway, a cloud server, and an SDN switch. Table 1 presents the details of the components of the testbed.

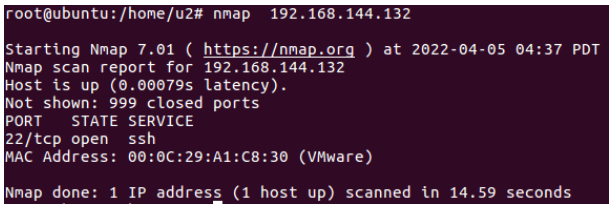
First, the components of the SDN network are deployed and run. The SDP gateway and the CTRL module running on the SDN controller begin to work, and finally, the cloud server and edge devices are connected to the network.

Table 1: Testbed configuration

machine	softwaree	detailed description
Edge device A +IH	-Linux Ubuntu16.04 -Fwknop module (IH)offered by Waverley Labs	Simulate the edge device in the wired network environment and run the SDP client IH model; directly connect to the SDN switch.
Edge device B +IH	-Linux Ubuntu16.04 -Fwknop module (IH) offered by Waverley Labs	Simulate the edge device in the wired network environment and run the SDP client IH model; directly connect to the SDN switch.
SDN controller +CTRL	-Linux Ubuntu16.04 -SDP controller module offered by Waverley Labs -OpenDaylight Boron	Simulate an SDN controller running the CTRL application to control the global view of the network; all edge devices must authenticate with CTRL before they can access cloud services.
SDP gateway +AH	-Linux Ubuntu16.04 -Fwknop module (AH) offered by Waverley Labs	Simulate the SDP gateway, run the SDP client AH model; configure the firewall rule to "deny all".
Cloud server	-Linux Ubuntu16.04	Simulate a cloud service hidden behind a gateway, and can perform basic SSH connection services.
SDN switch	-Linux Ubuntu16.04 -Open vSwitch 2.6	Simulate multiple SDN switches with data forwarding and processing capabilities, and establish topological relationships.



(a) With SDP



(b) Without SDP

Figure 5: Port scanning attack

4.2 Security Test

In the security test, we launched two types of attacks: 1) using the free nmap utility to launch a port scanning attack on the cloud server; 2) using the hping3 tool to launch a DoS attack on the cloud server, and using Wireshark to capture the traffic.

CTRL in SDP distributes the keys and certificates of edge devices. To simplify the operation during simulation, we randomly distribute keys and certificates for trusted edge devices manually. So for the port scanning part of the experiment, device A as an attacker, is not distributed keys and certificates. Device A performs port scanning attacks on cloud servers with and without SDP protection, respectively, and the results are shown in Figure 5(a)(b). The results show that when attacker A launches the attack without SDP protection, nmap scan shows that the TCP connection of the server is open. But under SDP protection, the device will not be granted access because the attacker is not CTRL authenticated. The result is as expected, and all ports are filtered.

In addition, from the 20s of capture, an SYN flood attack was launched against the network-protected service for 40 seconds. Wireshark was used to trace the captured traffic for the 80s, and the results are shown in Figure 6. The red curve is recorded as the traffic of attack packets, and the green curve represents the protected service traffic capture. CTRL instructs the SDN controller and the SDN switch to drop attack packets during the attack, and the server is not affected.

The results confirm that our proposed solution has "network stealth" properties, which can protect the server from unknown users.

4.3 Communication Reliability Test

For the cloud-edge environment of the power Internet of Things, ensuring real-time data transmission between the cloud and edge devices is the key for enterprises to efficiently provide services to users, such as remote control of smart homes and drone inspections of power equipment. Therefore, this paper uses the message delivery success rate (MDR) indicator to evaluate the communication reliability,

$$MDR = \frac{\sum R}{\sum S}. \tag{1}$$

where R is the number of messages received, and S is the number of messages sent. MDR is designed to calculate

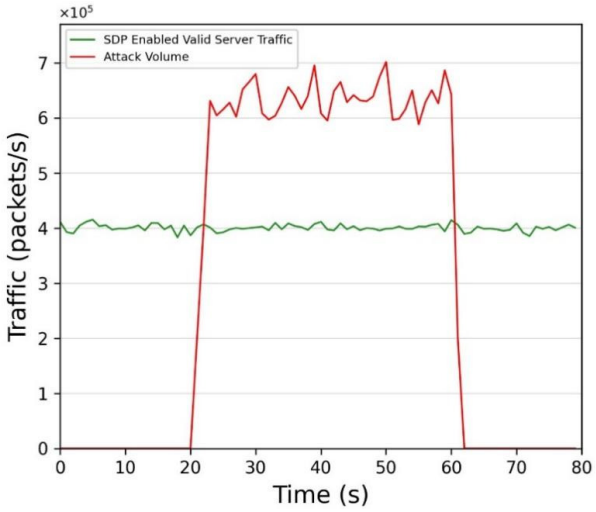


Figure 6: DoS attack capture using Wireshark.

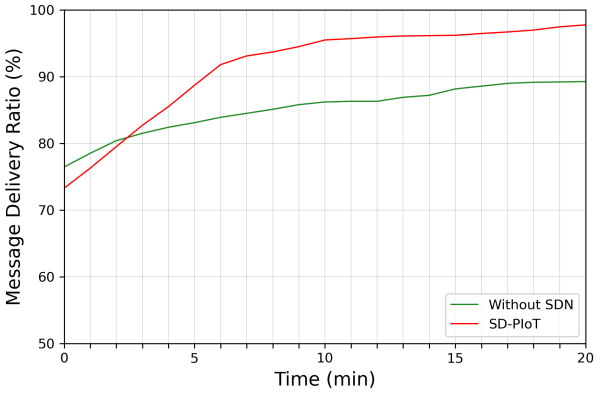


Figure 7: Comparison of MDRs

the ratio of the number of messages sent to the received in the network over some time. The closer the MDR value is to 1, the more reliable the communication.

In the simulation, trusted device A and device B send packets of 128 bits to the cloud server at regular intervals using wired and wireless connections. The cloud server responded immediately after receiving the data and returned the packet. And this process lasted for 20 minutes. To test the flexibility of this solution, we set the message sending interval of device A to 0.025s and device B to 0.05s, resulting in a relatively congested link between device A and the cloud server. Comparing the scheme with the SDN module and the scheme without the SDN module in this paper, the message delivery success rate is shown in Figure 7. It can be seen from Figure 7 that at the end of the simulation (20 minutes), the MDR of the proposed SD-PIoT framework achieves a success rate of 97.76%, while the performance of the scheme removing the SDN module drops by 8.5%. This is because the network congestion will increase the probability of packet loss. In this solution, the SDN controller is introduced to control the network topology accurately. The packet is forwarded in the optimal path and another redundant path under the condition of trusted identity. This can effectively avoid packet loss and improve MDR. The experimental results strongly confirm the effectiveness of this scheme for the network communication reliability of the massive terminal network in the power IoT.

4.4 Performance Test

To more objectively evaluate the performance of SD-PIoT, we compare the theoretical latency with the actual latency in the experiments. Ideally, ignoring processing and queuing delays, the total delay can be expressed as:

$$TD = \frac{M}{V} + \frac{L}{R}. \tag{2}$$

where M is the link length, V is the signal propagation speed, L is the packet length, R is the link bandwidth, and TD is designed to calculate the sum of transmission delay and propagation delay.

We monitored delays ten times using the Wireshark tool, analyzed packet exchanges between components, and averaged them to determine CTRL, switch, and gateway overhead. The statistical results are shown in Table 2. The latency of this scheme is higher than that of SDP-only and SDN-only. This is because edge devices must pass CTRL authentication and wait for the SDP gateway to update the firewall before accessing the server. In addition, during this period, the SDN controller will issue commands to instruct the SDN switch to add new flow entries. The SDN switch will search for matching flow entries when receiving packets from edge devices, inevitably increasing the delay. But from the results in the table, compared with the advantages of security, communication and network management brought by this framework, these time overheads are completely tolerable. In addition, the calculated theoretical value is not much different from the actual measured value, which verifies the correctness of our test.

4.5 Comparison of Security Features

To highlight our work's contribution, we summarize our project's features in this section. We have compared our scheme with other state-of-the-art schemes in terms of these characteristics. The results are shown in Table 3. This scheme improves the cloud-side security and communication reliability of the power IoT, ensures SDN security, and simplifies network management, which other schemes cannot achieve.

Table 2: Latency Analysis

Delay	Theoretical (secs)	Actual Measured (Sec)
End-to-End latency with SD-IoT	0.56184562	0.58145238
End-to-End latency without SDP	0.44853264	0.51365214
End-to-End latency without SDN	0.50124563	0.55856241
CTRL overhead	-	0.04512114
Switch overhead	-	0.02132412
Gateway overhead	-	0.04823564

Table 3: Features comparison

Features		**SD-PIoT**	**SDP-SDN [17]**	**Solution [20]**	**MEC-SDP [19]**
Service stealth		Y	N	Y	Y
communication reliability		Y	Y	Y	N
Network	Scalability	Y	Y	N	N
	Programmability	Y	Y	N	N
SDN inner layer security		Y	N	-	-

5 Conclusions

The in-depth integration of power IoT and cloud computing has led to various security challenges in the cloud-side environment. This paper proposes the SD-PIoT composition framework and illustrates its effectiveness in addressing these challenges. First, we discussed the security issues of the power IoT cloud-side environment and the difficulty of network management. We then describe how SDN and SDP can be cleverly combined to alleviate these challenges. Furthermore, we implement and test the proposed solution, demonstrating the superiority of the framework in security and network management. Tests and evaluations prove that the SD-PIoT framework can effectively resist DoS attacks and achieve "network stealth" of services. And the framework leverages the centralized programmability and scalability provided by SDN technology to control the network, allowing power IoT devices from heterogeneous networks to communicate securely and reliably while maintaining performance. In comparison with other schemes, the contribution of this paper can be reflected intuitively. Next, we will explore the implementation of this scheme in a more realistic power IoT scenario.

Acknowledgments

This work was supported in part by the State Grid Jiangxi Information & Telecommunication Company Project "Research on de-boundary security protection technology based on zero trust framework " under Grant 52183520007V.

References

[1] B. Bilger, "Software defined perimeter working group sdp specification 1.0," *Cloud Security Alliance, Tech. Rep*, vol. 4, 2014.

[2] M. Conti, P. Kaliyar, and C. Lal, "Censor: Cloud-enabled secure iot architecture over sdn paradigm," *Concurrency and Computation: Practice and Experience*, vol. 31, no. 8, p. e4978, 2019.

[3] CSA. "Software defined perimeter,". tech. rep., Cloud Security Alliance, http://downloads.cloudsecurityalliance.org/initiatives/sdp/SoftwareDefinedPerimeter.pdf, 2013.

[4] R. K. Das, N. Ahmed, F. H. Pohrmen, A. K. Maji, and G. Saha, "6le-sdn: an edge-based software-defined network for internet of things," *IEEE Internet of Things Journal*, vol. 7, no. 8, pp. 7725–7733, 2020.

[5] X. Xu et al., "Secure service offloading for internet of vehicles in sdn-enabled mobile edge computing," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 6, pp. 3720–3729, 2020.

[6] J. Garbis and J. Koilpillai, "Software defined perimeter architecture guide," *Cloud Security Alliance*, 2019.

[7] S. S. Jajoo and K. N. Kumar, "A review on deep-learning based network intrusion detection systems," *International Journal of Electronics and Information Engineering*, vol. 13, no. 4, pp. 170–179, 2021.

[8] Y. Jing. "Forecast of cloud computing market trend in 2022,". tech. rep., Tencent, http://cloud.tencent.com/developer/article/1919520?from=15425, Dec. 2021.

[9] R. Kaur and J. Kaur, "Cloud computing security issues and its solution: A review," in *2015 2nd International Conference on Computing for Sustainable Global Development (INDIACom)*, pp. 1198–1200. IEEE, 2015.

[10] H. Liang, H. Liu, F. Dang, L. Yan, and D. Li, "Information system security protection based on sdn technology in cloud computing environment," in *2021 IEEE International Conference on Advances in Electrical Engineering and Computer Applications (AEECA)*, pp. 432–435. IEEE, 2021.

[11] J. Ma, "Research on ddos in cloud computing environment (in chinese)," *China Computer & Communication*, pp. 149–151, 2017.

[12] A. Markandey, P. Dhamdhere, and Y. Gajmal, "Data access security in cloud computing: A review," in *2018 International Conference on Computing, Power and Communication Technologies (GUCON)*, pp. 633–636. IEEE, 2018.

[13] S. Meena, E. Daniel, and N. A. Vasanthi, "Survey on various data integrity attacks in cloud environment and the solutions," in *2013 International Conference on Circuits, Power and Computing Technologies (ICCPCT)*, pp. 1076–1081. IEEE, 2013.

[14] A. Moubayed, A. Refaey, and A. Shami, "Software-defined perimeter (sdp): State of the art secure solution for modern networks," *IEEE network*, vol. 33, no. 5, pp. 226–233, 2019.

[15] A. Refaey, A. Sallam, and A. Shami, "On iot applications: a proposed sdp framework for mqtt," *Electronics Letters*, vol. 55, no. 22, pp. 1201–1203, 2019.

[16] T. Ristenpart, E. Tromer, H. Shacham, and S. Savage, "Hey, you, get off of my cloud: exploring information leakage in third-party compute clouds," in *Proceedings of the 16th ACM conference on Computer and communications security*, pp. 199–212, 2009.

[17] A. Sallam, A. Refaey, and A. Shami, "On the security of sdn: A completed secure and scalable framework using the software-defined perimeter," *IEEE access*, vol. 7, pp. 146577–146587, 2019.

[18] S. N. Shirazi, A. Gouglidis, A. Farshad, and D. Hutchison, "The extended cloud: Review and analysis of mobile edge computing and fog from a security and resilience perspective," *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 11, pp. 2586–2595, 2017.

[19] J. Singh, Y. Bello, A. Refaey, A. Erbad, and A. Mohamed, "Hierarchical security paradigm for iot multiaccess edge computing," *IEEE Internet of Things Journal*, vol. 8, no. 7, pp. 5794–5805, 2020.

[20] J. Singh, A. Refaey, and J. Koilpillai, "Adoption of the software-defined perimeter (sdp) architecture for infrastructure as a service," *Canadian Journal of Electrical and Computer Engineering*, vol. 43, no. 4, pp. 357–363, 2020.

[21] J. Singh, A. Refaey, and A. Shami, "Multilevel security framework for nfv based on software defined perimeter," *IEEE Network*, vol. 34, no. 5, pp. 114–119, 2020.

[22] A. Sydney, *The evaluation of software defined networking for communication and control of cyber physical systems*. Kansas State University, 2013.

[23] T. Theodorou and L. Mamatas, "Sd-miot: A software-defined networking solution for mobile internet of things," *IEEE Internet of Things Journal*, vol. 8, no. 6, pp. 4604–4617, 2020.

[24] Z. Xiaojian, C. Liandong, F. Jie, W. Xiangqun, and W. Qi, "Power iot security protection architecture based on zero trust framework," in *2021 IEEE 5th International Conference on Cryptography, Security and Privacy (CSP)*, pp. 166–170. IEEE, 2021.

[25] C. H. Zhang, Y. Cui, H. Y. Tang, and J. P. Wu, "State-of-the-art survey on software-defined networking (sdn)," *Journal of software*, vol. 26, no. 1, pp. 62–81, 2015.

[26] B. Zhao, "Research on cloud computing security risk and security technology(in chinese)," *Computer Knowledge and Technology*, vol. 15, no. 2, pp. 27–28, 2019.

Biography

Rixuan Qiu is in State Grid Jiangxi Information & Telecommunication Company? Nanchang, China. Email: qiurixuanwork@163.com.

Yu Fu is in State Grid Jiangxi Information & Telecommunication Company? Nanchang, China. Email: gzgsgdfy@126.com.

Jian Le was born in Huanggang, Hubei, China in 1975. He received his Ph.D. degree in electrical engineering from Tsinghua University (THU), Beijing, China in 2006. He is currently Associate Professor with the college of electrical engineering and automation at Wuhan University (WHU), where he has been working on smart grid operation and power quality control technology. Email: lej01@tsinghua.org.cn.

Fuyong Zheng is in State Grid Jiangxi Information & Telecommunication Company, Nanchang, China. Email: 414833044@qq.com.

Gan Qi was born in Hengyang, Hunan, China in 1999. He received his B.S. degree from the School of Electrical Engineering and automation at Wuhan University (WHU), Wuhan, China, in 2021. He is now working towards a Master degree in electrical engineering at Wuhan University. He has been working on distribution generation control technology and power electronics.

Chao Peng is in State Grid Jiangxi Information & Telecommunication Company, Nanchang, China. Email: pengchao1988421@163.com.

Yuancheng Li was a postdoctoral research fellow in the Digital Media Lab, Beihang University. He has been with the North China Electric Power University, where he is

a professor and the Dean of the Institute of Smart Grid and Information Security. He was a postdoctoral research fellow in the Cyber Security Lab, college of information science and technology of Pennsylvania State University.

A Speech Fully Homomorphic Encryption Scheme for DGHV Based on Multithreading in Cloud Storage

Qiu-yu Zhang and Yu-gui Jia
(Corresponding author: Qiu-yu Zhang)

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

(Received Apr. 27, 2022; Revised and Accepted Oct. 12, 2022; First Online Oct. 15, 2022)

Abstract

Aiming at the problems of low encryption and decryption efficiency of existing speech homomorphic encryption schemes in cloud storage, significant expansion of speech ciphertext, and DGHV fully homomorphic encryption scheme is only a single-bit encryption form, a speech fully homomorphic encryption scheme for DGHV based on multithreading was proposed. Firstly, the proposed scheme converts floating point speech data to integer speech data by preprocessing. Secondly, the preprocessed speech data is disassembled into binary strings by the bits, and the disassembled binary speech data is converted into a matrix for cyclic encryption. Finally, the multithreading technology and homomorphic speech encryption scheme of many-to-one is used for parallel encryption to obtain the final ciphertext speech data. Performance analysis and experimental results show that the proposed scheme has high security, encryption, decryption efficiency, less ciphertext expansion, and can resist various conventional attacks.

Keywords: Cloud Storage; DGHV Fully Homomorphic Encryption; Multithreading; Speech Encryption

1 Introduction

With the continuous advancement of Internet and multimedia technologies, cloud computing has gained widespread attention and application due to its high flexibility, strong versatility, and support for massive information processing [15]. Since the cloud service provider is an untrustworthy third party, when the data is outsourced to the cloud service provider, the user will lose control of the data, especially to protect the privacy information contained in sensitive speech and other multimedia data [18, 28].

Currently, in order to protect user privacy and data security in the cloud, data owners usually choose to encrypt the data. Fully homomorphic encryption is a powerful encryption technique that can perform arbitrary computations on encrypted data, while mapping the corresponding computations to plaintext, the result of computations is still ciphertext. With this encryption technique, the data can be outsourced in encrypted form to any untrusted server to perform "ciphertext computation" to obtain the service, which ensures data security. In the meantime, this encryption technique also creates favorable conditions for the realization of ciphertext speech recognition and ciphertext speech retrieval [3].

Speech as the main carrier of multimedia data, which contains a lot of important content and confidential information especially in some special environments, such as confidential meetings, military activities, important cases, medical and health systems, etc. These sensitive information involving national level and enterprise confidentiality as well as personal privacy data should be given highly priority when storing. Although fully homomorphic encryption (FHE) technology ensures secure data storage and ciphertext computability, most of the existing FHE schemes are only for integer-type data types, whic makes it difficult to achieve infinite addition and multiplication operations, while security and encryption and decryption efficiency cannot be balanced. In recent years, most of the existing speech encryption schemes adopt chaotic encryption schemes, and there are relatively few research results on speech homomorphic encryption schemes, and the existing speech homomorphic encryption schemes are based on the use of partial homomorphic encryption (PHE), and there are fewer FHE methods for speech.

To address the above problems, in order to achieve efficient FHE of speech data in the cloud, based on the FHE scheme of DGHV (Dijk-Gentry-Halevi-Vaikuntanathan) [9] and using multithreading technology, a speech fully homomorphic encryption scheme for DGHV based on multithreading is proposed. The contributions of this work are as follows:

1) To address the problem that the DGHV scheme is only in the form of single-bit encryption, the preprocessed speech data is split into binary strings by bits, and the split binary speech data is converted into a matrix for cyclic encryption, which can realize encryption of unequal length speech data.

2) To address the problem of low encryption and decryption efficiency of DGHV scheme, multithreading technique is used to process speech data in parallel, which can make full use of CPU resources and effectively reduce the encryption time, and the computational complexity is reduced by batch processing technique.

3) To address the problem that the DGHV scheme is based on one-to-one (one encryption party and one decryption party) homomorphic encryption design, which is not applicable to the cloud environment and has high computational complexity, a model of speech homomorphic encryption scheme based on many-to-one (multiple encryption parties and one decryption party) is proposed and the correctness and homomorphism of the model is proved.

The organization of the paper is as follows: Section 2 deals with the related works. In Section 3, the theoretical knowledge related to the DGHV algorithm and the principle of multithreading technique is described in detail. The proposed scheme is elaborated in Section 4. The security and performance analysis and experimental results are given in Section 5. At last, there is the conclusion in Section 6.

2 Related Works

At present, existing speech encryption schemes include AES encryption, chaotic encryption, super chaotic encryption, RSA encryption, DES encryption and other methods, and these methods do not support ciphertext domain operations. In addition, since cloud servers are untrustworthy third parties that will cause data privacy leakage problems, While homomorphic encryption technology can solve the data confidentiality protection problem in outsourced computing such as cloud computing and prevent cloud computing service providers from accessing sensitive plaintext data [19]. Homomorphic encryption schemes can be classified into three categories: Partially homomorphic encryption (PHE) somewhat homomorphic encryption (SHE), and Fully homomorphic encryption (FHE). Among them, the FHE has broad application prospects. For example, cloud security, ciphertext retrieval/identification/authentication, encrypted databases, ciphertext interrogation for search engines, etc. In 2009, Gentry *et al.* [9] proposed the first FHE scheme based on ideal lattices, realizing the construction of "privacy homomorphism". In order to avoid the complex algebraic structure of the Gentry scheme based on the ideal lattice, Dijk *et al.* [8] proposed a FHE scheme of DGHV over integers, whose operations are based entirely on integer operations and are more easily understood. Mahmood *et al.* [17] proposed a hybrid homomorphic encryption scheme based on the GM (Goldwasser-Micali) encryption algorithm which is additively (single bit) homomorphic, and RSA algorithm which is multiplicative homomorphic. Naqvi *et al.* [21] proposed a multilayer PHE text steganography, robustness is achieved through implanting multilayer encoding concept where block encoding, key generation process is also improved by employing identical range for the selection of both prime numbers in PHE algorithm to generate public and private keys for encryption process. Cominetti *et al.* [6] proposed two efficient PHE schemes built upon the approximate common divisor problem, believed to be resistant to quantum computer attacks. In order to reduce the memory overhead of homomorphic encryption calculation. Chen *et al.* [4] proposed a hybrid SHE scheme that exploits the packing algorithm of the HEAAN scheme and the variant of the FV scheme, this scheme has smaller ciphertexts. Chillotti *et al.* [5] proposed a fast FHE scheme over the torus and bootstrapping circuits, bootstrapping circuits can speed up addition operations. In order to reduce the risk of data privacy leakage in the process of outsourced clustering. Jia *et al.* [13] proposed a privacy protection scheme of DBSCAN clustering based on homomorphic encryption.

In recent years, homomorphic encryption techniques are mostly for images, with relatively little research on speech data, and most existing homomorphic encryption schemes for speech are SHE schemes. For example, Ibtihal *et al.* [11] proposed a secure architecture composed by two clouds a private cloud dedicated for encryption/decryption, the encryption technique uses PHE and its homomorphism is verified using a watermarking algorithm. Yin *et al.* [29] proposed an improved elliptic curve cryptography by combining with homomorphic encryption, this new algorithm not only has good encryption effect, high security and large amount of key, but also has good sensitivity to initial value and anti-attack ability. Shankar *et al.* [24] proposed multiple key-based homomorphic encryption technique for image. For increasing the security level of encryption and decryption processes, the optimal key is selected using adaptive whale optimization algorithm. In order to solve the problem of the massive ciphertext speech data expansion and high computational complexity, Shi *et al.* [25] proposed a probabilistic statistics and addition homomorphic cryptosystem, comparing with Paillier homomorphic cryptosystem, the proposed scheme has less data expansion and lower computational complexity. Imran *et al.* [22] proposed an El-Gamal algorithm with asymmetric keys for speech, the cryptosystem performance is evaluated via different quality measures for audio signals encryption/decryption, and has robustness against effect.

Multithreading technology can use CPU resources efficiently, when a thread blocks, another thread will be executed to achieve parallel processing of data, thus re-

ducing overhead and improving the execution efficiency of the system [16]. In recent years, in order to improve the encryption and decryption efficiency of encryption systems, this technology has been widely used. For example, Gupta *et al.* [10] proposed RSA algorithm based on multithreading, and have enabled secure cloud storage of outsourced data. Al Essa *et al.* [1] proposed the use of multi-core processors to improve the performance of the AES algorithm, thus increasing the speed of encryption/decryption, using parallel design algorithms to improve the throughput. Jain *et al.* [12] applied multithreading strategies to neural network models, thus improving execution time. Cui *et al.* [7] designed a real-time data encryption system based on DES algorithm, which made full advantage of scheduling algorithms and multithreading CPU to improve the security and timeliness of data network communication.

In summary, existing homomorphic encryption schemes generally have low encryption and decryption efficiency, high computational complexity, and large ciphertext data expansion, and homomorphic encryption schemes are mostly applied to the image field, and relatively little research has been applied to homomorphic encryption of speech data. Therefore, this paper presents a speech FHE scheme for DGHV based on multithreading, which can make full use of CPU resources and improve the efficiency of encryption algorithm by using multithreading technology. The use of cyclic encryption can solve the problem that the DGHV scheme [7] can only encrypt single-bit plaintexts, and realize the encryption and decryption of unequal length speech.

3 Preliminaries

3.1 DGHV Encryption Algorithms

DGHV algorithm [8] is an asymmetric public key encryption algorithm whose security relies on the approximate GCD problem, and DGHV scheme [8] consists of four algorithms:

1) KeyGen(λ): Randomly generate a η bites odd integer p as the private key sk according to the security parameters, where $p \in (2Z+1)\bigcap[2^{\eta-1},2\eta)$,randomly select τ samples x_i, which x_i belongs to $D_{\gamma,\rho}(p)$ distribution (i=0,1,$\cdots$,τ).Relabel so that x_0 is largest odd and restart $r_p(x_0)$ is even, The public key is $\boldsymbol{pk}=(x_0, x_1, \cdots, x_\tau)$.

2) Enc($\boldsymbol{pk}$,$m\in\{0,1\}$): Chose a random set S $\subseteq\{1, 2,\cdots, \tau\}$, and a random integer r in $r\in(-2^{\rho'},2^{\rho})$, input plaintext $m\in\{0,1\}$, output ciphertext $c=\left[m+2r+\sum_{i\in S}x_i\right]_{x_0}$.

3) Evaluate(pk, C_ε, c_1, $\cdots$, c_t): Give the binary circuit C_ε with t inputs, and t ciphertexts c_i, apply the integer addition and multiplication gates of C_ε to the ciphertexts, performing all the operations over the integers, and return resulting integer.

4) Dec(sk, c): output plaintext m=(c mod p) mod 2.

After completing the above PHE scheme, then the bootstrapping technique is used to transform the PHE scheme into a FHE scheme, making it possible to perform calculations for circuits of arbitrary depth.

3.2 Multithreading and Thread Pool

Multithreading refers to the parallel processing of multiple threads from software or hardware [2], the purpose of multithreading is to synchronize multiple tasks and improve the efficiency of the system by increasing the resource usage. Multithreading is the application of the principle of concurrent execution mechanism to a program by dividing a program into several subtasks, and multiple subtasks are executed concurrently, each of which is a thread [16].

In this paper, the speech FHE scheme for DGHV uses multithreading technique, if encrypting speech data sequentially, firstly, request to encrypt the first frame of speech data and get the first frame of ciphertext speech data returned and store the first frame of ciphertext speech data; then request to encrypt the second frame of speech data. According to this implementation idea, a lot of time will be wasted on requesting and returning data. Therefore, using multithreading technique to request the second frame of speech data while waiting for the first frame of encrypted speech to return data, which can effectively improve the efficiency of speech encryption.

A thread pool is a form of multithreading processing. Using a thread pool can flexibly control the number of threads according to the system requirements and hardware environment, and all threads can be managed and controlled in a unified manner, thus improving the system operation efficiency. Figure 1 shows the processing flow chart of the thread pool.

4 The Proposed Scheme

4.1 System Model

Figure 2 shows the system model of the proposed speech FHE scheme for DGHV based on multithreading, the system model consists of 3 entities: data owner (DO), cloud server (CS), and data user (DU).

As shown in Figure 2, the specific responsibilities of the 3 entities in this model are as follows:

1) Data owner (DO): DO has a lot of local speech data that needs to be stored on CS. In the speech homomorphic encryption system of the proposed scheme, DO has the local speech data $D=\{D_1, D_2, \ldots, D_t\}$. In order to ensure the privacy and security of the speech data, firstly, multithreading DGHV fully homomorphic encryption is performed for this speech

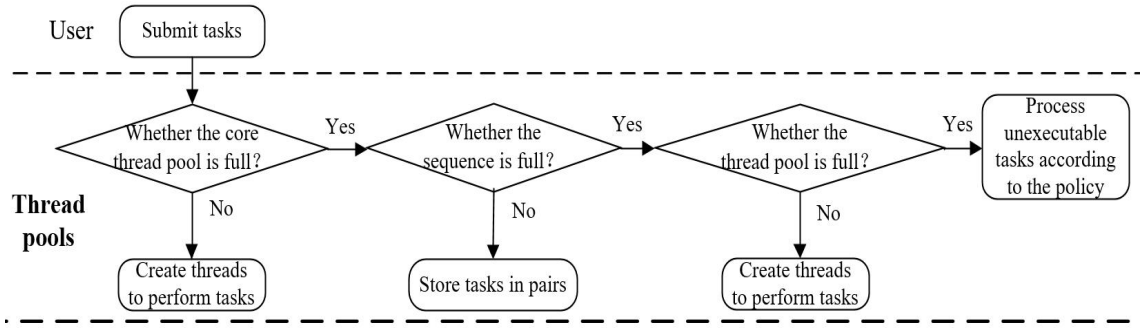


Figure 1: The processing flow chart of the thread pool.

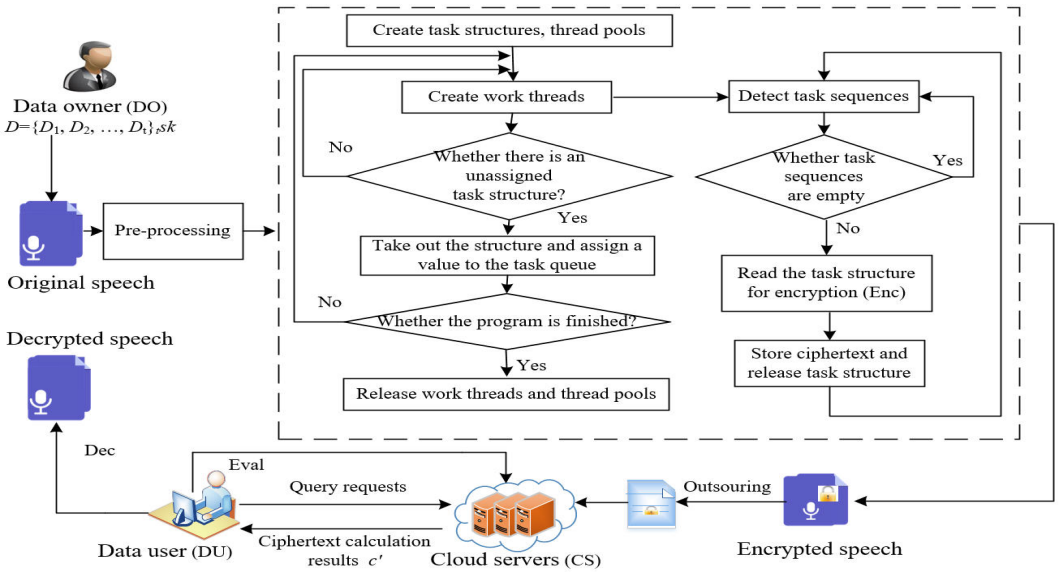


Figure 2: The system model of the proposed scheme.

data (Enc), then get the ciphertext speech data $c = \{c_1, c_2, \ldots, c_t\}$, which t is number of speech data, finally, the generated ciphertext speech data c is outsourced to CS for storage.

2) Cloud server (CS): CS provides ciphertext speech storage and performs ciphertext (Eval) operation. DO uploads ciphertext speech data c to CS for storage, CS performs a ciphertext operation to c, get the result of the ciphertext operation c'. When DU needs to send query requests to CS, CS returns the query result c' to DU.

3) Data user (DU): DU gets the ciphertext calculation result c' from CS, then use the private key sk decryption (Dec) provided by DO to get decrypted speech data.

4.2 Multithreading Algorithm Flow

Threads take up system resources when they are running, where T is the bit of time that it takes for a thread to complete a task request. T consists of three parts. Where T_1 is thread creation time, where T_2 is time for the thread to perform its task, including the time for synchronization between threads, where T_3 is thread destruction time. Multiple threads are managed by a thread pool as shown in Figure 1, the structure of threads directly affects the efficiency of the thread pool, to improve the encryption and decryption efficiency of speech DGHV fully homomorphic encryption scheme, Using multithreading technology, the encryption and decryption process can be performed simultaneously by multiple core processors to achieve parallelism and reduce the overall execution time of encryption and decryption. The specific process of the multithreading algorithm is as follows:

1) During the request process, threads are generally executed in the order in which the speech files arrive, with those that arrive first being processed first and those that arrive later waiting for the previous speech files to complete before being executed in turn. To reduce the thread waiting time, all speech files are sorted by file size, which threads with fewer speech

files are executed first and those with larger speech files are executed later.

2) The sorted threads are divided into two groups, with the large speech files running in a fixed thread pool and the small speech files running in a temporary thread pool, which the grouping strategy can reduce the occupation of system resources by threads in a short period of time and improve the running efficiency of the whole program.

3) When executing small speech files, set the thread pool size larger to improve the running efficiency of the thread pool. When executing threads for large speech files, the large speech file is split.

4) The pre-processed speech file is split into m copies of the same size data stream, the size of the split speech file is F_i/m_i, and the split speech file is stored using data table. Before the split, when the speech file F_i is requested by a thread, the time to complete the request of that thread is t_i. While splitting the speech file into m copies, the split speech file is requested with m threads and the time required for the completed F_i is t_i/m. After splitting the speech file, the running time of each thread becomes $1/m$ of the original.

5) The split speech file will be encrypted request using threads, and to prevent deadlocks in the process of storing the speech file in threads, each thread uses a mutual exclusion lock.

6) When the speech data in the thread completes the request, the speech recorded in the data table is deleted in order to save system resources.

4.3 Encryption and Decryption Scheme Construction

The speech multithreading DGHV fully homomorphic encryption scheme designed in this paper is based on the DGHV scheme [21] and use multithreading techniques to improve the efficiency of encryption and decryption. Figure 3 shows the speech multithreading DGHV fully homomorphic encryption and decryption processing flow.

The specific processing of the speech multithreading DGHV fully homomorphic encryption scheme is as follows:

1) Sampling, quantization and analog-digital conversion. Firstly, the original speech signal is sampled and quantized, and then the sampled and quantized analog speech signal is converted into a digital speech signal.

2) Speech data pre-processing. The digital speech signal is processed by pre-weighting, framing and windowing.

3) Positive integer processing. To enable homomorphic encryption of speech data and to improve the efficiency of encryption, it is necessary to convert floating-point speech data into positive integer speech data. Encodes floating-point speech data by multiplying it by a large number with fixed precision and rounding the result, and decodes it by dividing by this large number. By multiplying the original speech data of floating-point type by 10^{φ}, keeping φ bits valid, where φ is 6.

4) The processed positive integer speech data is disassembled into binary string by bit, and the disassembled speech data is formed into a matrix for cyclic encryption.

5) Speech multithreading DGHV fully homomorphic encryption. The proposed scheme contains 3 algorithms: Key generation algorithm (KeyGen(λ)), Encryption algorithm (Enc(sk, $\boldsymbol{pk}$, m)), ciphertext calculation algorithm (Eval($\boldsymbol{pk}$, C, c_1, c_2, $\cdots$, c_t)), multithreading processing details are detailed in Section 4.2. The specific processing process is as follows:

Step 1. Key generation algorithm(KeyGen(λ)). Randomly generate a η bites large prime number p as private key,where $p\in[2^{\eta-1},2\eta]$, Using randomly selected se initialization to get a pseudo-random generator $f(se)$, Randomly select a non-square 2^{λ}-rough integer q_0, where $q_0\in[0,\ 2^{\gamma}/p]$,Make encryption modulus $x_0=q_0\times p$. Generate public key base volume $\mathcal{X}_{i,b}\in[0,x_0]$ by using $f(se)$, then use Equation (1) to calculate the public key offset $\delta_{i,b}$.

$$\delta_{i,b}=[\chi_{i,b}]_p+\xi_{i,b}\cdot p-r_{i,b} \tag{1}$$

where $i\le\beta$, $0\le b\le 3$, $r_{i,b}\in(-2^{\rho},2^{\rho})$, $\xi_{i,b}\in[0,2^{\gamma+\eta}/p]$, make $x_{i,b}=\mathcal{X}_{i,b}$-$\delta_{i,b}$, then the public key vector $\boldsymbol{pk}=(se,\ x_0(\delta_{i,b}),\ 1\le i\le\beta,\ 0\le b\le 3)$.

Step 2. Encryption algorithm (Enc(sk, $\boldsymbol{pk}$, m)). Firstly, use $f(se)$ to recover the integers $\mathcal{X}_{i,b}$, calculate $x_{i,b}$ by using Equation (2), then choose a random vector of integer coefficients $\boldsymbol{b}=\boldsymbol{b}_{i,j,k,l}$, where $1\le i,j,k,l\le\beta$ and $\boldsymbol{b}_{i,j,k,l}\in[0,2^{\alpha})$, and then output ciphertext

$$c=\left[m+2r+2\sum_{1\le i,j,k,l\le\beta}b_{i,j,k,l}\cdot x_{i,0}\cdot x_{j,1}\cdot x_{k,2}\cdot x_{l,3}\right]_{x_0}$$

$$x_{i,b}=\chi_{i,b}-\delta_{i,b} \tag{2}$$

Step 3. Ciphertext calculation algorithm (Eval($\boldsymbol{pk}$, C, c_1, c_2, $\cdots$, c_t)). Input t ciphertexts c_i, $1\le i\le t$, and the public key $\boldsymbol{pk}$ to an arithmetic circuit C_{ϵ} with t input ports, perform addition and multiplication operations and get the ciphertext result c^*.

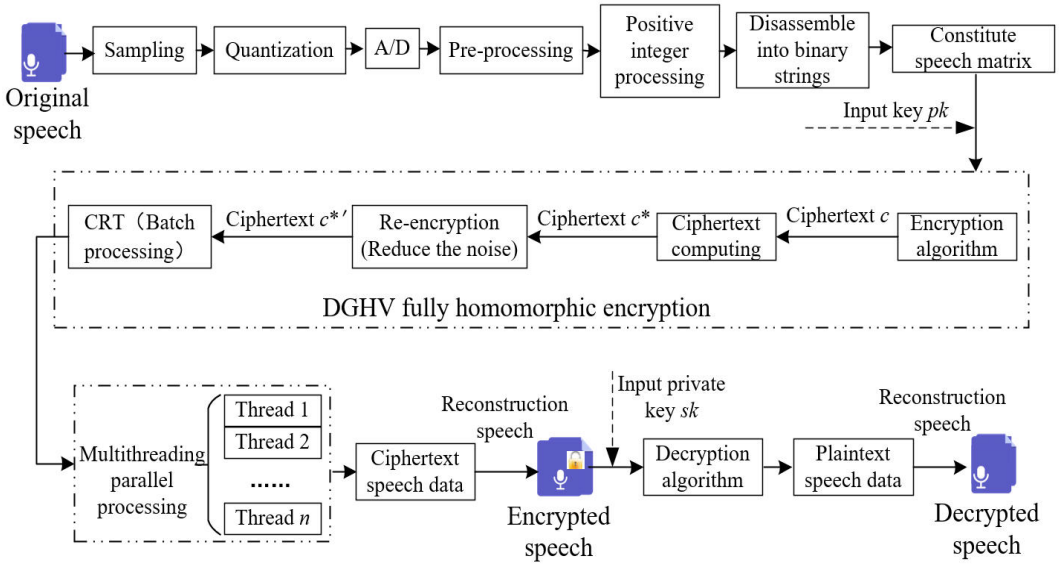


Figure 3: Speech multithreading DGHV fully homomorphic encryption and decryption processing flow.

Step 4. Re-encryption. Assume there are two plaintexts m_1 and m_2, firstly, encryption operation is performed using the above speech fully homomorphic encryption scheme, get ciphertext c_1 and c_2,and next the new ciphertext is obtained by multiplication $c_3(c_3{=}c_1{\times}c_2)$, then the ciphertext c_3 is ciphertext refreshed using re-encryption technique, finally get the ciphertext c_4, the generated ciphertext c_4 can be used for the next step of ciphertext homomorphism calculation, after re-encryption technology can reduce the noise generated by multiplication operations.

Step 5. Since the DGHV algorithm can only encrypt single-bit plaintexts, Chinese remainder theorem (CRT) messages containing multiple bits can be encrypted as a ciphertext, convert the above encryption scheme into a batch DGHV fully homomorphic encryption scheme by using CRT, the form of the ciphertext processed in batches using CRT is as in Equation (3).

$$c = CRT_{q0,p0,\ldots,pt-1}(q, 2r_0 + m_0, \cdots, 2r_{l-1} + m_{l-1}) \quad (3)$$

6) Reconstruction speech.

Step 1. Through the inverse process of data migration and data expansion, the above ciphertext speech data is recovered as floating point speech data.

Step 2. Reorder and scale of floating point speech data.

Step 3. Set the appropriate number of channels, quantization bits, sampling frequency and sampling points, recover ciphertext speech data to speech.

7) Speech decryption.Input private key sk and ciphertext calculation results c^*, use $m'{=}[c^* \bmod p] \bmod 2$ and get plaintext speech data m', plaintext speech data through reconstruction speech to get decrypted speech.

4.3.1 Ciphertext Noise Analysis

Ciphertext noise in homomorphic encryption increases in a variety of ways, the source of noise is the value of initial noise after one or more homomorphic operations between two or more ciphertexts, the main forms of noise for the DGHV fully homomorphic encryption scheme are exponential multiplication, constant addition and linear addition.

For exponential multiplication of noise: Input two ciphertext speech data c_1 and c_2, after the homomorphic multiplication operation the ciphertext c_3 is obtained,the noise in ciphertext c_3 is $b''{=}2^n b$, after a homomorphic operation the noise is 2ρ, after many times of the same homomorphic operations, the noise will increase exponentially with the number of multiplications in the form of $2^n\rho$. To prevent decryption failure, the modulus p should be set to kp, the multiplication operation can be performed $\log_2 k$ several times.

For constant addition noise: Input two ciphertext speech data c_1 and c_2, after the homomorphic multiplication operation the ciphertext c_3 is obtained, If $\sum_{i=1}^{n} r_i$ satisfy $\sum_{i=1}^{n} r_i < p/2$, then you can get the correct decrypted data.

For linear addition of noise: Multiplication of cipher-

text adds noise by adding a constant value, the growth of the noise is independent of the number of multiplications of the ciphertext.

4.3.2 Ciphertext Expansion Analysis

There are 4 security parameters in the multithreading speech DGHV fully homomorphic encryption scheme, Which are toy safety parameter, small safety parameter, medium parameter and large safety parameter, where λ, ρ, η, γ, β and α are this parameter specifies the values of the four security parameters,the specific values are shown in Table 1.

Table 1: Parameters of multithreading speech DGHV fully homomorphic encryption scheme

Parameters	λ	ρ	η	γ	β	α
Toy	42	26	988	2.9×10^6	8	936
Small	52	41	1558	1.6×10^7	11	1476
Medium	62	56	2128	8.5×10^7	17	2016
Large	72	71	2698	3.9×10^8	25	2556

The proposed scheme reduces the key size by decreasing the number of public key elements in the public key vector and decreasing the bit length of each public key element. According to the specific values taken in Table 1, calculate the amount of public key storage for the proposed scheme, the size of the public key corresponding to the four security parameters is 100KB.

4.4 Many-to-One Speech FHE Scheme

The existing speech homomorphic encryption schemes are all for one-to-one homomorphic encryption and decryption on integers, with high computing complexity, resulting in low encryption and decryption efficiency, which cannot be applied to the scenarios of big data and cloud environment. A many-to-one homomorphic encryption model is one that contains multiple encryption parties $P_i(i=1, 2, ..., n)$ and one decryption party, The parameters of the many-to-one speech homomorphic encryption scheme are the same as those in Section 4.3.2, the structure model of many-to-one speech FHE scheme is shown in Figure 4.

The many-to-one speech FHE scheme includes 3 algorithms, each specific algorithm is described as follows:

1) Key generation algorithm KeyGen(λ): The many-to-one based speech encryption scheme consists of multiple encryption parties $P_i(i=1, 2, ..., m)$ and one decryption party P. The public key of the decryption party P is the public key vector $\boldsymbol{pk}=(se, x_0(\delta_{i,b}), 1\le i\le\beta, 0\le b\le3)$ generated in Section 4.3. The private key sk_i of the encryption party $P_i(i=1, 2, ..., m)$: Firstly, randomly choose a large prime number of η_i bits as p_i as the private key sk_i, where $p_i\in[2^{\eta_i-1}, 2^{\eta_i}]$, P_i performs a random vector substitution on the public key of P, and get $\boldsymbol{pk}=(se, x_0, \boldsymbol{\delta}_{1,b}, \boldsymbol{\delta}_{2,b}, \cdots, \boldsymbol{\delta}_{3,b}), 0\le b\le3)$, next randomly select integers $q_{i,0}$ and $q_{i,1,b}, \cdots, q_{i,\beta,b}$, and random noises $r_{i,0}$ and $r_{i,1,b}, \cdots, r_{i,\beta,b}$, where $q_{i,0}$ and $q_{i,1,b}, \cdots, q_{i,\beta,b}\in[0,2^{\gamma_i}]/p_i)$ and $r_{i,0}$ and $r_{i,1,b}, \cdots, r_{i,\beta,b}\in(-2^{\rho_i},2^{\rho_i})$, then the encryption modulus $x_{i,v,b}$ can be calculated by Equation (4).

$$x_{i,v,b}=\chi_{i,v,b}-\delta_{i,v,b}=p_i\cdot q_{i,v,b}(\chi_{v,b}-\delta_{v,b})+2r_{i,v,b} \tag{4}$$

where $1\le v\le\beta_i, 0\le b\le3$.

Make $x=(\mathcal{X}_{v,b}$ - $\boldsymbol{\delta}_{v,b})$, and then Equation (4) is converted to Equation (5):

$$x_{i,0}=p_iq_{i,0}x_0+2r_{i,0} \tag{5}$$

Make $x_{i,0}$ in Equation (5) be the maximum, then the public key of the encryption party P_i is shown in Equation (6).

$$pk_i=(se, x_0, x_{i,0}, \delta_{i,1,0}, \cdots, \delta_{i,1,3}, \delta_{i,\beta_i,1}, \delta_{i,\beta_i,3}) \tag{6}$$

2) Encryption algorithm Enc(m, $\boldsymbol{pk}_i$): The encryption party P_i randomly selects the integer vector $\boldsymbol{\mu}_i=\boldsymbol{\mu}_{i,v}, 1\le v\le\beta_i$ and random integers $k_i\in(-2\rho'_1, 2\rho'_1)$, the output ciphertext c_i is shown in Equation (7).

$$c_i=[m+2k_i+2\sum_{1\le v_1,v_2,v_3,v_4\le\beta}\mu_{i,v}\cdot x_{i,v_1,0}\cdot x_{i,v_2,1}\cdot x_{i,v3,2}\cdot x_{i,v_4,3}]_{x_{i,0}} \tag{7}$$

3) Decryption algorithm Dec(c_i): The encryption party P_i inputs the private key sk_i, also $sk_i=p_i$ then the decrypted speech data is obtained as in Equation (8). The decryption party P inputs the private key sk, also $sk=p$, then the decrypted speech data is obtained as in Equation (9).

$$m_i \leftarrow [c_i \bmod p_i]_2 \tag{8}$$

$$m_i \leftarrow [c_i \bmod p]_2 \tag{9}$$

4.4.1 Proof of Correctness

The correctness of the many-to-one speech FHE scheme will be proved in 3 parts in the following: the correct decryption of the ciphertext by the decryption party P, the encryption party P_i can use the private key sk_i for correct decryption, and the decryption party P can use private key $sk=p$ for correct decryption.

1) Proof of correct decryption by the decryption party: The key pairs (sk, $\boldsymbol{pk}$) of the decrypting party P are generated by the encryption scheme in Section 4.3, the proposed scheme is able to decrypt ciphertext speech data correctly, therefore the decryption party can also decrypt the ciphertext speech data.

2) The encryption party P_i can use the private key $sk_i=p_i$ to prove correct decryption: The ciphertext is obtained by Equation (7), and maintain $x_{i,0}$ the maximum, then there exists an integer N_i that converts

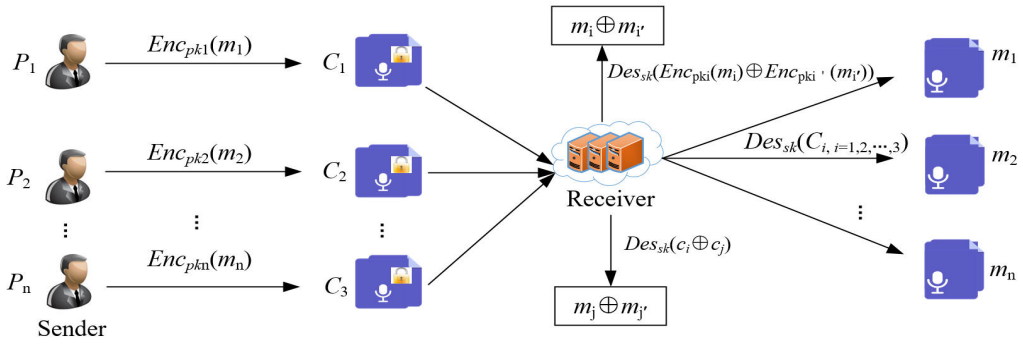


Figure 4: The structure model of many-to-one speech FHE scheme.

Equation (7) to Equation (10).

$$\begin{aligned} & c_i = m_i + 2k_i + N_i x_{i,0} \\ & +2 \sum_{1\le v1,v2,v3,v4\le\beta} \mu_{i,v} \cdot x_{i,v_1,0} \cdot x_{i,v_2,1} \cdot x_{i,v_3,2} \cdot x_{i,v_4,3} \\ & = m_i + 2k_i + 1 + N_i \cdot \left(p_i q_{i,0} x_0 + 2r_{i,0}\right) \\ & +2 \sum_{1\le v1,v2,v3,v4\le\beta} \mu_{i,v} \cdot \left(p_i \cdot q_{i,v,b}\left(\chi_{v,b} - \delta_{v,b}\right) + 2r_{i,v,b}\right) \end{aligned} \tag{10}$$

Make $D = k_i + \sum_{1\le v_1,v_2,v_3,v_4\le\beta} \mu_{i,v} \cdot 2r_{i,v,b} + N_i r_{i,0}$, $L = 2 \sum_{1\le v_1,v_2,v_3,v_4\le\beta} \mu_{i,v} \cdot q_{i,v,b}\left(\chi_{v,b} - \delta_{v,b}\right) + N_i q_{i,0} x_0$, Then Equation (10) is transformed into Equation (11).

$$c_i = m_i + 2D + p_i L \tag{11}$$

According to the safety parameters in Table 1, it can be seen that $|m_i + 2D| < P$, then it means $m_i = (c_i \bmod p_i) \bmod 2$, the encryption party P_i is able to decrypt correctly with the private key $sk_i{=}p_i$.

3) The decryption party P uses the private key $sk{=}\ p$ to prove correct decryption: Transform Equation (9) into Equation (11), then the ciphertext is Equation (11).

From Section 4.3, we know that $(\mathcal{X}_{v,b}\text{-}\boldsymbol{\delta}_{v,b}) \bmod p = r_{v,b}$, and exist integer $t_{v,b}$, make $\mathcal{X}_{v,b}\text{-}\boldsymbol{\delta}_{v,b}{=}r_{v,b}{+}t_{v,b}p$, which $x_0{=}q_0 p$, substitute into Equation (10) to obtain Equation (12).

$$c_i = m_i + 2G + pF \tag{12}$$

where $G = s_i + \left(k_i + \sum_{1\le v1,v2,v3,v4\le\beta} \mu_{i,v} \cdot 2r_{i,v,b} + N_i r_{i,0}\right)$, $F = p_i \cdot \left(2 \sum_{1\le v1,v2,v3,v4\le\beta} \mu_{i,v} q_{i,v,b}\left(\chi_{v,b} - \delta_{v,b}\right) + N_i q_{i,0} x_0\right)$.

According to the safety parameters in Table 1, it can be seen that $|m_i{+}2\mathrm{G}|{<}P$, then it means $m_i = (c_i \bmod p_i) \bmod 2$, the decryption party P is able to decrypt correctly with the private key $sk{=}p$.

4.4.2 Proof of Homomorphism

In the following, the homomorphism of the many-to-one speech homomorphic encryption scheme will be proved in two parts: the homomorphism proof of the decryption party P, the homomorphism proof of the encryption party P_i.

1) Proof of homomorphism of the decryption party P: The key pairs (sk, $\boldsymbol{pk}$) of the decryption party P are generated by the encryption scheme in Section 4.3, the proposed scheme satisfies additive and multiplicative homomorphism, Thus the decryption party P also satisfies the additive and multiplicative homomorphism.

2) Proof of homomorphism of the encryption party P_i: Assume there are two speech data $s_{i,1}$ and $s_{i,2}$, the ciphertext speech data of the encryption party can be obtained by Equation (7): respectively $c_{i,1}{=}s_{i,1}{+}2D_1{+}p_i L_1$, $c_{i,2}{=}s_{i,2}{+}2D_2{+}p_i L_2$.

   a. The additive homomorphism is Equation (13):

$$\begin{aligned} & ((c_{i,1} + c_{i,2}) \bmod p_i) \bmod 2 \\ & = ((s_{i,1} + 2D_1 + p_i L_1 + s_{i,2} + 2D_2 + p_i L_2) \bmod p_i) \\ & \quad \bmod 2 \\ & = ((s_{i,1} + s_{i,2} + 2(D_1 + D_2) + p_i(L_1 + L_2) \bmod p_i) \\ & \quad \bmod 2 \\ & = s_{i,1} + s_{i,2} \end{aligned} \tag{13}$$

   b. The multiplicative homomorphism is Equation (14):

$$\begin{aligned} & ((s_{i,1} s_{i,2}) \bmod p) \bmod 2 \\ & = ((s_{i,1} + 2D_1 + pL_1)(s_{i,2} + 2D_2 + pL_2) \bmod p) \\ & \quad \bmod 2 \\ & = \left(\begin{array}{l} s_{i,1} s_{i,2} + 2\left(s_{i,1} D_2 + s_{i,2} D_1 + 2D_1 D_2\right) + \\ p\left(s_{i,1} L_2 + 2D_1 L_2 + L_1\left(s_{i,2} + 2D_2 + pL_2\right)\right) \end{array} \right) \\ & \quad \bmod p \bmod 2 \\ & = s_{i,1} s_{i,2} \end{aligned} \tag{14}$$

   To sum up the above, both additive homomorphism and multiplicative homomorphism are proved, it shows that the encryption party Pi

satisfies the homomorphism to the ciphertext speech data.

5 Experimental Results and Performance Analysis

In order to evaluate the performance of the proposed scheme, this paper mainly analyzes both theoretical and experimental aspects, and the dataset is selected from the open Chinese speech database THCHS-30 [27] of Tsinghua University, sampling frequency of 16kHz, single channel wav format speech with 16bit sampling accuracy. Different types of speech are selected and intercepted for 2s, 4s, 6s, 8s and 10s as the data of this experiment as shown in Table 2. The experimental hardware platform is Intel(R) Core(TM) i5-4200H CPU, Memory 16GB. Software Environment: Windows 10, JetBrains PyCharm Community Edition 2020.1 x64, MATLABR2017b software implementation simulation.

Table 2: Experimental speech type

Speech file	Speech length (s)	Speech format
A11_2.wav	2	wav
B2_329.wav	4	wav
A2_0.wav	6	wav
C4_579.wav	8	wav
D32_987.wav	10	wav

5.1 Performance Analysis of Encryption Scheme

5.1.1 Key Space Analysis

The key space of the DGHV homomorphic encryption scheme is the set of integers, the key parameters of the proposed scheme are designed as: The key length L is 37, the key size is $L\times16$=592, the key space size is 2^{592}, According to the Big Bang theory, the universe has been around for 224 millennia, And the time required to search for a 2^{128} bit key is 1.46×10^{15} years. Therefore, the proposed scheme has higher security, can resist exhaustive attacks, there are no security problems such as speech data leakage and malicious tampering.

5.1.2 Histogram Analysis

Histograms [27] are widely used to evaluate the quality of speech signals because of their objectivity and visualization, a better performing encryption system with a uniformly distributed histogram of the encrypted speech amplitude. Taking the speech of A2_0.wav in Table 2 as an example. The histogram of the amplitude of the original and encrypted speech is shown in Figure 5.

As can be seen from Figure 5, the original speech amplitude histogram in Figure 5(a) is unevenly distributed and has irregular statistical characteristics. As shown in Figure 5(b), the histogram of encrypted speech amplitude is basically uniformly distributed, without large ups and downs and fluctuations. Therefore, the proposed scheme has better masking ability for the statistical characteristics of speech data. By observing Figure 5(a) and Figure 5(c), it can be seen that there is no difference between the amplitude histograms of the original speech and the decrypted speech, indicating that the proposed scheme can achieve lossless recovery of the encrypted speech signal.

5.1.3 Signal-to-Noise Ratio and Segmented Signal-to-Noise Ratio

Audio signal-to-noise ratio (SNR) [22] is the ratio of the normal sound signal strength to the noise signal strength, for the encrypted speech signal, it refers to the ratio of the original speech signal to the encrypted signal, This metric measures the noise and distortion of the encrypted speech signal. The SNR is calculated by Equation (15).

$$SNR = 10\log_{10}\frac{\sum_{i=0}^{L} x_i^2}{\sum_{i=0}^{L}[x_i - y_i]^2} \tag{15}$$

The segmented signal-to-noise ratio (SNRseg) [22] is the average of the short-frame SNR values, this value is one of the most common indicators used to evaluate the quality of speech signal, the formula for calculating the SNRseg is shown in Equation (16).

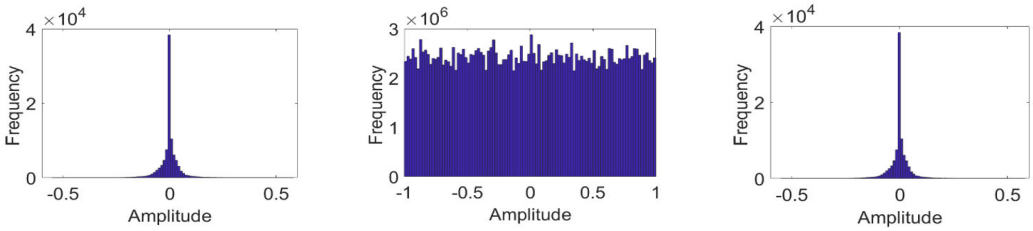
$$SNRseg = \frac{10}{M}\times\sum_{m=0}^{M-1}\log_{10}\frac{\sum_{n=Lm}^{Lm+L-1} x_i^2}{\sum_{n=Lm}^{Lm+L-1}[x_i - y_i]} \tag{16}$$

where x_i is the original speech signal, y_i is the encrypted speech signal, L denotes the number of samples, and M is the number of frames in the speech signal.

Table 3: SNR and SNRseg of encrypted speech signals

speech file	SNR (dB)	SNRseg (dB)
A11_2.wav	-149.8764	-121.1328
B2_329.wav	-157.0241	-119.5891
A2_0.wav	-148.1735	-127.1212
C4_579.wav	-156.5644	-119.5891
D32_987.wav	-146.8266	-135.3196
Average value	-151.693	-124.5504

As can be seen from Table 3, The SNR and SNRseg of the encrypted speech signal in the proposed scheme are low, which can enough to satisfy the requirements of speech encryption indexes, indicating that the encrypted speech in the proposed scheme is completely noise, and it is difficult for the attacker to obtain any information. Therefore, the proposed scheme has a high level of security.



(a) Original speech (b) Encrypted speech (c) Decrypted speech

Figure 5: Histogram of original and encrypted speech amplitude.

5.1.4 Correlation Analysis

Correlation coefficient [25] is a simple and easy analytical method to measure the relationship situation between quantitative data and is widely used to evaluate the encryption performance. The range of the correlation coefficient is [-1, 1], If the correlation coefficient of two speech signals is between -1 and +1, then it means that the two speech signals are strongly correlated; If the correlation coefficient of two speech signals is around 0, then it means that the correlation between these two speech signals is weak. The correlation coefficients are calculated as in Equation (17) to Equation (20).

$$r_{ml} = \frac{C\,(m,l)}{\sqrt{\sigma\,(m)}\sqrt{\sigma\,(l)}} \tag{17}$$

$$C\,(m,l) = \frac{\sum (m-\overline{m})(l-\bar{l})}{N-1} \tag{18}$$

$$\sigma\,(m) = \frac{1}{N}\sum_{i=1}^{N}(m_i-\overline{m})^2 \tag{19}$$

$$\overline{m} = \frac{1}{N}\sum_{i=1}^{N} m_i \tag{20}$$

where $\overline{m}$ and $\bar{l}$ re respectively the mean values of the original speech signal m and the encrypted speech signal l, $C(m,\ l)$ is the covariance between the original speech signal m and the encrypted speech signal l, $\sigma(m)$ and $\sigma(l)$ denote the variance between the original speech m and the encrypted speech l.

Taking the speech of A2_0.wav in Table 2 as an example, the waveforms of the original speech and the encrypted speech are shown in Figure 6. Figure 6(a), Figure 6(b), and Figure 6(c) respectively show the original speech, encrypted speech, and decrypted speech waveforms.

From Figure 6(b), it can be seen that the waveform graph of the encrypted speech signal does not have any speech waveform characteristics, which indicates that the encryption effect of the proposed scheme is better. In order to judge the performance of the encryption algorithm of the proposed scheme against statistical analysis attacks, the analysis is performed by calculating the correlation coefficient of the encrypted speech signals. The Pearson correlation coefficient of speech signals is calculated using Equation (17) to measure the correlation between speech signals, The calculation results are shown in Table 4.

Table 4: Speech correlation analysis

Speech file	Ori/Enc	Ori/Dec
A11_2.wav	0.0034	0.9899
B2_329.wav	0.0039	0.9999
A2_0.wav	0.0028	0.9925
C4_579.wav	0.0025	0.9962
D32_987.wav	0.0019	0.9942
Average value	0.0029	0.9943

As can be seen from Table 4, the correlation coefficient between original speech and encrypted speech is near 0, which mean that there is no correlation between original speech and encrypted speech, which indicates that the encryption performance of the encryption algorithm of the proposed scheme is better. The correlation coefficient between the original speech and the decrypted speech is between -1 and +1, which indicates that the decryption algorithm of the proposed scheme can basically achieve the lossless recovery of speech data.

5.1.5 Information Entropy Analysis

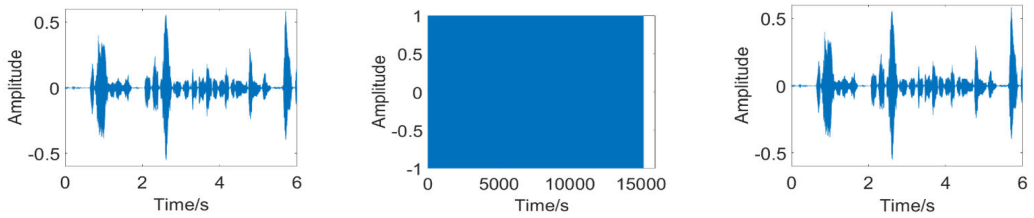
Information entropy [25] can be used to measure the "confusion level" of encrypted messages, if the information entropy of encrypted data is higher, the attacker gets the less raw data, which indicates the better the encryption. The calculation method of information entropy is shown in Equation (21):

$$H(k) = -\sum_{k=0}^{S} p(k)\log_2 p(k) \tag{21}$$

where $p(k)$ is the input speech data. S represents the number of sampling points.

The closer the information entropy of the encrypted speech data is to 16, which indicates the better the encryption performance of the speech encryption system and the higher the security. Table 5 shows the information entropy of encryption and decryption speech data.

As can be seen from Table 5, the entropy of encrypted speech is close to 16, which indicates that the proposed scheme has high security and can resist entropy attacks.



(a) Original speech (b) Encrypted speech (c) Decrypted speech

Figure 6: Waveform of original speech signal and encryption and decryption speech signal.

Table 5: The information entropy of encryption and decryption speech data

Speech file	Original speech	Encryption speech
A11_2.wav	12.3532	15.5546
B2_329.wav	12.0468	15.6244
A2_0.wav	11.7500	15.6935
C4_579.wav	12.4650	15.6697
D32_987.wav	12.2345	15.5548
Average value	12.1700	15.6194

5.1.6 Analysis of Resistance to Differential Attacks

Number of Changing Pixel Rate (NPCR) is an important metric to evaluate the resistance to differential attacks and is widely used in the field of speech data encryption [14], the closer the value of NPCR is to 100% and can resist differential attacks, which indicates the better the encryption performance of the encryption scheme. The NPCR values for different speech are shown in Table 6.

Table 6: NPCR for different speech

Speech file	NPCR (%)
A11_2.wav	100.00
B2_329.wav	99.86
A2_0.wav	99.92
C4_579.wav	99.90
D32_987.wav	99.94
Average value	99.99

As can be seen from Table 6, the value of NPCR of the proposed scheme is close to 100%, indicating that the encrypted speech data is completely different from the original speech data, which indicates that the proposed scheme is sufficient to resist differential attacks.

5.1.7 Efficiency Analysis

The security and encryption efficiency of speech encryption system balance each other, when the security of encryption algorithm is high, and the efficiency of encryption and decryption is low, which is not suitable for encryption and decryption of large amount of speech data. Table 7 shows the efficiency analysis of multithreading and serial encryption and decryption.

As can be seen from Table 7, it can be seen that using multithreading mode, the time required to encrypt 1s of speech in the proposed scheme is 1.7357s, and the time required to decrypt 1s of speech is 0.8222s; while using serial mode, the time required to encrypt 1s of speech is 27.7833s, and the time required to decrypt 1s of speech is 13.1658s. Analysis of the above data shows that using multithreading mode is better than serial mode, Because the multithreading processing mode can make full use of the advantages of multi-core processors to achieve parallel processing of speech encryption and decryption, set a larger number of threads for small speech files, perform multithreading parallel processing after splitting processing for large speech files, and use the Cupy computational library [26] to improve the computational power, therefore, the encryption and decryption efficiency of the proposed scheme is higher.

5.2 Performance Comparison Analysis with Existing Encryption Schemes

In order to objectively evaluate the encryption performance of the proposed scheme, experimental results were compared with existing speech chaos encryption schemes [20, 23] and speech homomorphic encryption schemes [14, 22, 25], All encryption schemes were used to select 4s of speech data from the TIMIT (Texas Instrument and Massachusetts Institute of Technology) [30] speech dataset for comparative analysis. Table 8 shows the comparison between the proposed scheme and different speech encryption schemes.

As can be seen from Table 8, the key size and key space of the proposed scheme are higher than those of the [14,20,22,23,25]. The larger the key space, the higher the security of the encryption algorithm against exhaustive attacks. Because the proposed scheme has 4 levels of security parameters, the key size and key space are higher. The SNR values of the proposed scheme and those of the [20, 22] are negative, but the values of proposed scheme are higher than those of the [20, 22], which indicates that the encrypted speech of the proposed scheme is completely noisy, then an attacker cannot obtain any speech information from this speech noise, further which indicates that the proposed scheme can effectively ensure the privacy of speech data. The SNRseg value of

Table 7: Efficiency analysis of multithreading and serial encryption and decryption

Speech file name	Speech Length (s)	Multithreading mode		Serial mode	
		Encryption (s)	Decryption (s)	Encryption (s)	Decryption (s)
A11_2.wav	2	3.3493	1.6892	53.6522	27.1234
B2_329.wav	4	7.0479	3.4386	112.8552	55.1276
A2_0.wav	6	10.8974	5.2749	174.4562	84.4084
C4_579.wav	8	13.5461	6.7395	216.8376	107.832
D32_987.wav	10	17.2311	7.5240	275.6976	120.484
Average	-	1.7357	0.8222	27.7833	13.1658

Table 8: Comparison between the proposed scheme and different speech encryption schemes

Evaluation Indicators	Ref. [25]	Ref. [22]	Ref. [14]	Ref. [20]	Ref. [23]	Proposed
Key size	512	512	512	239	214	592
key space	2^{512}	2^{512}	2^{512}	2^{239}	2^{214}	2^{592}
SNR (dB)	Inf	-35.0224	-	-13.6118	123.8900	-152.4907
SNRseg (dB)	Inf	-38.0201	-	-	122.8300	-118.4460
NPCR(%)	-	-	0.997	0.999	0.999	0.999
Cov Ori&Enc	0.0746	-	0.0004	0.5545	0.0104	0.0032
Cov Ori&Dec	1.0000	1.0000	0.8006	-	0.9981	1.0000
Enc time (s)	25.3075	-	1.1240	4.000	-	7.1456
Dec time (s)	24.2481	-	0.6240	-	-	3.5345
Ciphertext expansion	6.667×10^6	4.375×10^4	5.7×10^4	-	-	2.251×10^4
Resistance to differential attacks	×	×	✓	✓	✓	✓
Resistance to statistical attacks	✓	✓	✓	✓	✓	✓
Entropy attacks	×	×	×	×	×	✓

the proposed scheme is higher than the SNRseg of the scheme in the [22], which indicates that the decrypted speech obtained by the proposed scheme is of high quality; However, the SNRseg value of the proposed scheme is slightly lower than that of the [23], which is caused by the difference in the number of subframes to the speech signal. The NPCR values of the proposed scheme and the schemes in the [14, 20, 23] are basically the same, indicating that both the proposed scheme and the schemes in the comparative schemes are resistant to differential attacks. The Cov (original & encrypted) values of the proposed schemes are all lower than the Cov (original & encrypted) values of the schemes in the [20, 23, 25], and the original speech and encrypted speech in this paper are closer to 0, which indicates that the security of the proposed scheme is higher than the comparative schemes, and it is difficult for an attacker to obtain any speech information. The correlation between the original speech and the decrypted speech is stronger and the value is closer to 1, the quality of decrypted speech is higher. The Cov (original & decryption) values of the proposed scheme are all higher than those of the scheme in [14, 23], which are the same as those in [14, 23], which indicates that the decrypted speech obtained by the proposed scheme is basically distortion-free and can achieve lossless recovery. The encryption time and decryption time of the proposed scheme are better than the encryption and decryption time of the scheme in [25], since the proposed scheme processes floating-point speech data into integer speech data and uses multithreading technology to achieve parallel processing of encryption operations, the encryption and decryption time is improved; The encryption time and decryption time of the proposed scheme are lower than the encryption and decryption time of the scheme in [14], because the proposed scheme is a FHE scheme, the encryption and decryption times are lower than those in the [14]. Compared with the chaotic encryption scheme proposed in [20, 23], the encryption time of the proposed scheme is higher than that of [20, 23], however, all the remaining evaluation metrics are better than the chaotic encryption schemes of the [20, 23]. The proposed scheme not only ensures the security of cloud computing, but also supports ciphertext operation, which provides a good feasibility for the following practical applications such as speech retrieval / recognition / authentication in ciphertext domain. In addition, our scheme is able to resist conventional attacks such as differential attacks, statistical attacks, and entropy attacks compared to the comparative schemes.

6 Conclusions

In order to improve the encryption and decryption efficiency of the existing speech homomorphic encryption scheme, improve the existing DGHV full homomorphic encryption scheme can only carry out single-bit encryp-

tion, and the existing one-to-one encryption scheme is not suitable for cloud environment and big data scenarios. Using the advantage that multithreading technology can achieve parallel processing of encryption operations, a speech multithreading DGHV fully homomorphic encryption scheme is proposed based on the existing DGHV scheme, and a many-to-one speech FHE scheme model is designed. The main advantages of our scheme can be seen as follow: 1) Enables encryption of speech data of different lengths; 2) A many-to-one speech homomorphic encryption model is constructed using the proposed speech multithreading DGHV fully homomorphic encryption scheme, and the correctness and homomorphism of the model are proved; 3) The encryption and decryption efficiency of different speech length is analyzed by measuring the SNR, SNRseg and correlation coefficient, the proposed scheme can resist various conventional attacks such as differential attacks, statistical attacks and entropy attacks; 4) which can effectively improve the efficiency of the existing DGHV fully homomorphic encryption, reduce the amount of ciphertext expansion, and has higher security. It can be applied to the secure storage and ciphertext computation of massive ciphertext speech data.

Acknowledgments

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

References

[1] H. A. Al Essa and A. S. Ashoor, "Enhancing performance of aes algorithm using concurrency and multithreading," *ARPN Journal of Engineering and Applied Sciences*, vol. 14, no. 11, pp. 2039–2049, 2019.

[2] S. Aljawarneh, M. B. Yassein, W. A. Talafha, "A multithreaded programming approach for multimedia big data: encryption system," *Multimedia Tools and Applications*, vol. 77, no. 9, pp. 10997–11016, 2018.

[3] V. Biksham and D. Vasumathi, "A lightweight fully homomorphic encryption scheme for cloud security," *International Journal of Information and Computer Security*, vol. 13, no. 3-4, pp. 357–371, 2020.

[4] H. Chen, I. Iliashenko and K. Laine, "When heaan meets fv: a new somewhat homomorphic encryption with reduced memory overhead," in *IMA International Conference on Cryptography and Coding*, Springer, Cham, pp. 265–285, Dec 2021.

[5] I. Chillotti, N. Gama, M. Georgieva, *et al.*, "TFHE: fast fully homomorphic encryption over the torus," *Journal of Cryptology*, vol. 33, no, 1, pp. 34–91, 2020.

[6] E. L. Cominetti and M. A. Simplicio, "Fast additive partially homomorphic encryption from the approximate common divisor problem," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 2988–2998, 2020.

[7] A. Cui, H. Zhao, X. Zhang, *et al.*, "Power system real time data encryption system based on DES algorithm," in *2021 13th International Conference on Measuring Technology and Mechatronics Automation (ICMTMA)*, IEEE, Beihai, China. pp. 220–228, Jan 2021.

[8] M. V. Dijk, C. Gentry, S. Halevi, and V. Vaikuntanathan, "Fully homomorphic encryption over the integers," in *Annual international conference on the theory and applications of cryptographic techniques*, Springer, Berlin, Heidelberg, pp. 24–43, May 2010.

[9] C. Gentry, "Fully homomorphic encryption using ideal lattices," in *Proceedings of the forty-first annual ACM symposium on Theory of computing*, ACM, New York, pp. 169–178, May 2009.

[10] P. Gupta, D. K. Verma, A. K. Singh, "Improving rsa algorithm using multi-threading model for outsourced data security in cloud storage," in *2018 8th International Conference on Cloud Computing, Data Science and Engineering (Confluence)*, IEEE, Noida, pp. 14–15, Jan 2018.

[11] M. Ibtihal, E. O. Driss and N. Hassan, "Homomorphic encryption as a service for outsourced images in mobile cloud computing environment," *International Journal of Cloud Applications and Computing*, vol. 7, no. 2, pp. 27–40, 2017.

[12] N. Jain, K. Nandakumar, N. Ratha, *et al.*, "Optimizing homomorphic encryption based secure image analytics," in *2021 IEEE 23rd International Workshop on Multimedia Signal Processing (MMSP)*, IEEE, Tampere, pp. 1–6, Oct. 2021.

[13] C. Jia, R. Li and Y. Wang, "Privacy protection scheme of dbscan clustering based on homomorphic encryption," *Journal on Communications*, vol. 7, no. 2, pp. 27–40, 2021.

[14] M. S. Khoirom, D. S. Laiphrakpam and T. Tuithung, "Audio encryption using ameliorated ElGamal public key encryption over finite field," *Wireless Personal Communications*, vol. 117, no. 2, pp. 809–823, 2021.

[15] K. Lee, "Comments on "secure data sharing in cloud computing using revocable-storage identity-based encryption"," *IEEE Transactions on Cloud Computing*, vol. 8, no. 4, pp. 1299–1300, 2020.

[16] D. Liang, Y. Zhang, Z. Xu, *et al.*, "Pythagorean fuzzy Bonferroni mean aggregation operator and its accelerative calculating algorithm with the multithreading," *International Journal of Intelligent Systems*, vol. 33, no. 3, pp. 615–633, 2018.

[17] Z. H. Mahmood and M. K. Ibrahem, "New fully homomorphic encryption scheme based on multistage partial homomorphic encryption applied in cloud computing," in *2018 1st Annual International Conference on Information and Sciences (AICIS)*, IEEE, Fallujah, pp. 182–186, Nov 2018.

[18] A. Mohammad, "Distributed authentication and authorization models in cloud computing systems: A

literature review," *Journal of Cybersecurity and Privacy*, vol. 2, no. 1, pp. 107–123, 2022.

[19] S. J. Mohammed and D. B. Taha, "From cloud computing security towards homomorphic encryption: A comprehensive review," *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, vol. 19, no. 4, pp. 1152–1161, 2021.

[20] R. Nagakrishnan and A. Revathi, "A robust cryptosystem to enhance the security in speech based person authentication," *Multimedia Tools and Applications*, vol. 79, no. 29, pp. 20795–20819, 2020.

[21] N. Naqvi, A. T. Abbasi, R. Hussain, *et al.*, "Multilayer partially homomorphic encryption text steganography (mlphe-ts): a zero steganography approach," *Wireless Personal Communications*, vol. 103, no. 2, pp. 1563–1585, 2018.

[22] O. A. Imran, S. F. Yousif, I. S. Hameed, *et al.*, "Implementation of el-gamal algorithm for speech signals encryption and decryption," *Procedia Computer Science*, vol. 167, pp. 1028–1037, 2020.

[23] P. Sathiyamurthi and S. Ramakrishnan, "Speech encryption using hybrid-hyper chaotic system and binary masking technique," *Multimedia Tools and Applications*, vol. 81, no. 5, pp. 6331–6349, 2022.

[24] K. Shankar, S. K. Lakshmanaprabu, D. Gupta, *et al.*, "Adaptive optimal multi key based encryption for digital image security," *Concurrency and Computation: Practice and Experience*, vol. 32, no. 4, p. e5122, 2020.

[25] C. Shi, H. Wang, Y. Hu, *et al.*, "A speech homomorphic encryption scheme with less data expansion in cloud computing," *KSII Transactions on Internet and Information Systems (TIIS)*, vol. 13, no. 5, pp. 2588–2609, 2019.

[26] S. Tokui, R. Okuta, T. Akiba, *et al.*, "Chainer: A deep learning framework for accelerating the research cycle," in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ACM, pp. 2002–2011, Jul. 2019.

[27] D. Wang and X. Zhang, "Thchs-30: A free chinese speech corpus," *arXiv preprint* arXiv: 1512.01882, 2015.

[28] Y. Wu, X. Wang, W. Susilo, *et al.*, "Efficient server-aided secure two-party computation in heterogeneous mobile cloud computing," *IEEE Transactions on Dependable and Secure Computing*, vol. 18, no. 6, pp. 2820–2834, 2020.

[29] S. Yin, J. Liu and L. Teng, "Improved elliptic curve cryptography with homomorphic encryption for medical image encryption," *International Journal of Network Security*, vol. 22, no. 3, pp. 421–426, 2020.

[30] V. Zue, S. Seneff and J. Glass, "Speech database development at MIT: TIMIT and beyond," *Speech communication*, vol. 9, no. 4, pp. 351–356, 1990.

Biography

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

Yu-gui Jia is currently a master student of the School of Computer and Communication, Lanzhou University of Technology, China. She received the BS degrees in communication engineering from Tarim University, Xinjiang, China, in 2019. Her research interests include network and information security, audio signal processing and application, multimedia data security.

Tag Group Coexistence Protocol for Verifiable RFID System

Yu-Zhen Li[1], Dao-Wei Liu[2], and Wen-Tao Zuo[2]
(Corresponding author: Yu-Zhen Li)
School of Computer Science and Engineering, Hunan University of Information Technology[1]
Maotang Industrial Park, Changsha Economic and Technological Development Zone, Hunan 410151, China
Email: liyuzh90@163.com
Engineering College, Guangzhou College of Technology and Business, China[2]
Guangzhou 510006, China
(Received Apr. 29, 2022; Revised and Accepted Oct. 12, 2022; First Online Oct. 15, 2022)

Abstract

This paper proposes a protocol to prove the coexistence of tag groups. The protocol uses ergodic XOR operation to implement the encryption of the information to be transmitted; Ergodic XOR operation makes full use of the Hamming weight value carried by the encrypted private information, which can not only reduce the introduction of parameters and the storage space but also increase the difficulty of protocol cracking. At first, the steps of the proof protocol for the coexistence of single tag groups are given, and then an extensible proof protocol for the coexistence of multiple tag groups is supplied. From the comparative analysis of security and performance for different protocols, it can be shown that the proposed protocol can not only meet the security needs of users but also outperform other protocols in performance; It also can be shown that the protocol has rigorous logical reasoning by reasoning the protocol from the perspective of formalization.

Keywords: Internet of Things; RFID System; Tag Group; Coexistence Proof; Traversal XOR Operation

1 Introduction

After entering the new century, with the popularization and application of the Internet of things, RFID technology, as a key technology in the Internet of things (IoT), has been widely promoted and applied [4,11]. Typical radio frequency identification (RFID) systems include electronic tags, readers and background servers. Because their electronic tags have many advantages, such as small volume, easy to carry and deploy, low cost, long service time and so on, they are widely used in various fields. With the popularization and application of this technology, there are some security problems, such as the need to prove the coexistence of multiple electronic tags at one time to prevent the security problems caused by the counterfeiting of legal electronic tags by third parties [7,8].

Group Proof Protocol (GPP) is proposed to solve the coexistence problem of multiple electronic tags at a certain time. A typical set of examples of GPP is the application in medicine. On the doctors' medicine lists providing for patients, they not only require the very basic information of the medicine, but also require the precautions related to the use of these medicine. It's evidently that only one electronic tag is not enough to complete the tag information, the very need of at least two electronic tags gradually exist. One electronic tag is used to mark the basic information about one medicine, such as its name. Another electronic tag is used to mark the precautions of taking the drug, such as how much to take at a time. At the same time, in order to ensure the reliable safety of patients using this medicine, we need to ensure that encapsulate in a drug instructions must be the instructions on the use of the medicine, not the use of other medicines, tag where you need to make sure that any time these two information of electronic tags are was in a state of coexistence, otherwise the patients may be in health risks [6,13,14].

GPP can be divided into two categories according to the different roles of background servers in the use of GPPs, namely, online GPP and offline GPP [5]. Online GPP means that the background server needs to be online during the whole process of GPP. The offline GPP means that during the whole process of the GPP, the background server is only online at some times and offline at most times. Obviously, the efficiency of GPP in offline state is higher than that in online state, so most of the currently designed GPP belong to offline GPP. From the point of view of electronic tag communication, GPP can be divided into two categories: serial mode and parallel mode. Serial mode means that multiple electronic tags communicate with the background server in a certain order. The parallel mode means that multiple electronic

tags can communicate with the background server at the same time. Obviously, the efficiency in the parallel mode is much higher than that in the serial mode [3, 16]. This paper designs a GPP with good scalability in order to solve the defects of insufficient security and poor scalability in the design process of GPP.

The chapters in this paper are arranged as follows: Section one introduces the background knowledge of the research in this paper. The second Section introduces the advantages and disadvantages of GPP. In Section 3, the concrete implementation of traversal XOR operation is given. Section 4 introduces the design and implementation of the ultra-lightweight GPP in detail. Section 5 analyzes the security of GPP from the direction of multiple attack types. Section 6 analyzes protocol performance from the perspective of computation. In Section 7, GNY logic formalization is adopted to formalize the GPP. Section 8 gives the concrete implementation steps of the scalability of the GPP. Section 9 summarizes the work of the whole paper.

2 Related Research Works

In reference [9], an efficient GPP is designed. For protocol analysis, the protocol has ideal time complexity, but there are security defects in the protocol. Attackers can analyze useful privacy information based on the obtained messages, making the protocol unable to provide privacy security. In reference [10], a GPP is designed based on elliptic ECC, which can provide good security requirements. However, ECC is used to realize information encryption, so the amount of computation on the electronic tag side of the protocol is too large, which cannot be promoted and used in the electronic tag with low cost and limited computation.

Proposed in reference [2] a group can be cross resistance against proof protocol, protocol can resist complex crisscross, provides better security, but deal with the method of random Numbers, and other important information clear text, makes the attacker easy to get a random number, coupled with other get messages, but poor cite other part privacy information, makes the users' information confidential. In reference [1], a lightweight GPP is designed, which has an acceptable computation time complex. However, for protocol security analysis, electronic tags directly use part of the session message as the next-round authentication key, which makes the protocol have security risks. Attackers can eavesdrop and obtain the session message, and then know the next-round session key.

In reference [15], an off-line GPP is designed, which has good expansibility, but cannot provide forward security, that is, an attacker can analyze the next session message based on the current communication message, so that the attacker can communicate by impersonating one of the session entities. Reference [12] proposes a GPP with forward security, which has low computation time complexity and can be applied to electronic tags where low-cost computation is limited. However, the protocol cannot resist the exhaustive attack initiated by a third party, and the third party can enumerate possible values of private information by means of exhaustive attack.

Based on the above, considering that most of the existing GPPs have disadvantages such as large computation, security defects or inability to prove the coexistence of multiple tags, an off-line GPP is designed in this paper. This protocol can be used in the case of single tag grouping proof or multi-tag grouping proof. The protocol uses the traversal XOR operation with ultra-lightweight computation to encrypt information, which can greatly reduce the computation on the end of the electronic tag, making the protocol has good popularization and practicability.

3 Related Knowledge

Traversal XOR operation ($TXO(X,Y)$) defined as follows:X, Y, and Z are binary strings of L bits, P_x and P_y represent pointers to binary string X and binary string Y, H_x and H_y epresent hamming weights of binary string X and binary string Y, respectively. When $H_x \geq H_y$, pointers P_x and P_y are traversed from the first bit of binary string x and binary string y respectively. When pointer P_x traverses to the ith bit of binary string X is 0, the ith bit of binary string X and the ith bit of binary string Y will perform XOR operation. And places the result of the XOR operation in the ith bit of the binary string Z; When the pointer P_x traverses to the ith bit of the binary string X is 1, the ith bit of the binary string Y is directly placed at the ith bit of the binary string Z. When $H_x < H_y$, the Pointers P_x and P_y respectively from a binary string of binary string X,Y first began to traverse, when P_y points to traversal of binary string Y the ith the bit is 1, the ith of binary string X and the ith of binary string Y will exclusive or operation, and to place the results of an exclusive or operation on the ith of binary string Z; When the pointer P_y traverses the ith bit pointing to the binary string Y as 0, it directly places the number on the ith bit of the binary string X at the ith bit of the binary string Z.

4 RFID Tag Group Certification Protocol

This section first introduces the symbol meaning of the proof protocol in detail, and then gives the specific steps of the protocol in combination with the flowchart of the proof protocol.

1) Prove the meaning of the protocol symbol R stands for reader (the reader is equipped with powerful computing and query capabilities);

   T_A indicates the electronic tag A;

   T_B indicates the electronic tag B;

K_A represents the shared secret value between T_A and R;

K_B represents the shared secret value between T_B and R;

ID_A Indicates the identifier of the electronic tag A;

ID_{A-R} indicates the right half of the identifier of the electronic tag A;

ID_{A-L} indicates the left half of the identifier of the electronic tag A;

ID_B Indicates the identifier of the electronic tag B;

ID_{B-R} indicates the right half of the identifier of the electronic tag B;

ID_{B-L} indicates the left half of the identifier of the electronic tag B;

x_R represents the random number generated by R;

x_A represents the random number generated by T_A;

x_B represents the random number generated by T_B;

$R_i\,(i = 1, 2, 3, \cdots)$ represents the communication message generated by R calculation;

$A_{1i}\,(i = 1, 2, 3, \cdots)$ represents the communication message generated by T_A calculation;

$B_{1i}\,(i = 1, 2, 3, \cdots)$ represents the communication message generated by T_B calculation;

ACK indicates session request instruction;

$\oplus$ stands for XOR operation;

& represents and operations;

$TXO(X, Y)$ represents traversal XOR operation.

2) Implementation steps of GPP

The GPP consists of the initialization phase and the proof phase. In the initialization phase, the reader and tag information is set, and in the proof phase, the tag coexistence proof is completed.

After the initialization phase is complete, the R end of the reader stores K_A, K_B, ID_A, and ID_B. The electronic tag T_A stores information K_A and ID_A.electronic tag T_B stores information K_B and ID_B.

The flow chart of GPP based on traversal XOR operation is shown in Figure 1.

Combined with Figure 1, the steps of the ultra-lightweight tag group coexistence proof protocol in the paper can be described as follows:

Step 1. Reader R sends an ACK message to the electronic tag T_A to enable the tag group coexistence proof protocol.

Step 2. After receiving the message, the electronic tag T_A generates random number x_A and calculates the session message A_{11} and A_{12} in sequence. Finally, the electronic tag T replies the ID_{A-R},A_{11} and A_{12} message to the reader R. $A_{11} = x_A \oplus ID_{A-L}$,$A_{12} = TXO(x_A, ID_{A-L})$.

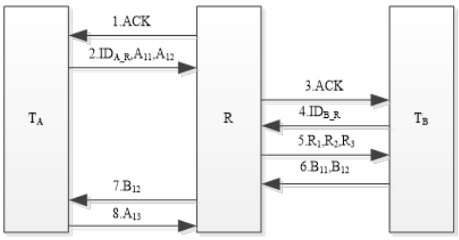


Figure 1: Flow chart of tag group coexistence certification protocol

Step 3. After receiving the message, the reader R looks for data equal to the received ID_{A-R} in the stored information.

Not found. GPP stopped.

If yes, the responding electronic tag is the one to be verified. Proceed to the next step. The reader R takes out the ID_{A-R} corresponding to ID_{A-L} and obtains A random number x_A through calculation. Then, the random number x_A calculated is combined with other parameter information according to the same algorithm to obtain A_{12}, and then compares whether A_{12} and A_{12} are the same.

The message source fails to pass reader R authentication and the GPP stops.

If they are the same, it indicates that they have passed the verification and $x_A = x_A$. Finally, the reader R sends an ACK message to the electronic tag T_B.

$$\begin{aligned} A_{12} &= TXO(x_A, ID_{A-L}) \\ &= TXO(A_{11} \oplus ID_{A-L} ID_{A-L}), \\ x_A &= A_{11} \oplus ID_{A-L}. \end{aligned}$$

Step 4. After receiving the message, the electronic tag T_B replies an ID_{B-R} message to the reader R.

Step 5. After receiving the message, the reader R looks for data equal to the received ID_{B-R} in the stored information.

Not found. GPP stops.

If yes, the responding electronic tag is the one to be verified. Proceed to the next step. The reader R takes out the ID_{B-L} corresponding to the ID_{B-R}, generates a random number x_R, calculates the session message R_1, R_2, R_3, and finally replies the message R_1, R_2, R_3 to the electronic tag T_B.

$$\begin{aligned} R_1 &= x_R \oplus ID_{B-L}, \\ R_2 &= x_A \oplus ID_{B-L}, \\ R_3 &= TXO(x_R, x_A). \end{aligned}$$

Step 6. After receiving the message, the electronic tag T_B calculates the random number $x_R = R_1 \oplus ID_{B-L}$ and $x_A = R_2 \oplus ID_{B-L}$ respectively. Then, the calculated random number x_A and x_R are combined with other parameter information to calculate R_3 according to the same algorithm, and then compare whether R_3 and R_3 are the same.

The message source fails to pass electronic tag T_B authentication, GPP stops.

If the two are the same, the verification is passed, and $x_A = x_A$ and $x_R = x_R$. Then the electronic tag T_B generates a random number x_B, calculates the session message B_{11} and B_{12} in turn, and finally replies the message B_{11} and B_{12} to the reader R.

$$\begin{aligned} R_3 &= TXO(x_R, x_A) \\ &= TXO(R_1 \oplus ID_{B-L}, R_2 \oplus ID_{B-L}), \\ B_{11} &= x_B \oplus (x_R \& x_A), \\ B_{12} &= TXO(x_B, x_B \oplus K_B). \end{aligned}$$

Step 7. After receiving the message, the reader R can obtain the random number $x_B = B_{11} \oplus (x_R \& x_A)$ through calculation, and then calculate the random number x_B based on other parameter information according to the same algorithm to get B_{12}, and then compare whether B_{12} and B_{12} are the same.

The message source fails to pass reader R authentication, GPP stops.

If they are the same, it indicates that they have passed the verification and $x_B = x_B$. The reader R then forwards the B_{12} message to the electronic tag T_A.

$$\begin{aligned} B_{12} &= TXO(x_B, x_B \oplus K_B) \\ &= TXO(B_{11} \oplus (x_R \& x_A), \\ &\quad (B_{11} \oplus (x_R \& x_A)) \oplus K_B). \end{aligned}$$

Step 8. After receiving the message, the electronic tag T_A calculates session message A_{13} using B_{12} and other parameters, and finally replies message A_{13} to reader R.

$$A_{13} = TXO(B_{12} \& x_A, K_A)$$

.

Step 9. After R receives the message, it calculates A_{13} according to the same algorithm and compares whether A_{13} and A_{13} are the same.

The message source fails to pass reader R authentication, GPP stops. If they are the same, the authentication succeeds, and the electronic tag T_A and electronic tag T_B are in the coexistence state.

$$\begin{aligned} A_{13} &= TXO(B_{12} \& x_A, K_A) \\ &= TXO(B_{12} \& (A_{11} \oplus ID_{A-L}), K_A). \end{aligned}$$

5 Safety

This section will analyze the protocols in this paper from different perspectives.

Impersonation attacks: Attacker lack of electronic tag identification information, the lack of Shared key between electronic tag and to read and write, so the attacker can calculate the correct session message $A_{1i}\,(i = 1, 2, 3, \cdots)$ or $B_{1i}\,(i = 1, 2, 3, \cdots)$, then the attacker can only randomly selected parameters as the unknown parameter values for operation, operation income session messages with the correct session value must be different, to read/write device after receiving the attacker sends to the message, A simple verification can identify the forged message from the attacker, and the impersonation attack fails.

Replay attack: An attacker can obtain round i session messages by eavesdropping on round i sessions. During round $i+1$ sessions, the attacker replay round i session messages obtained by eavesdropping in an attempt to obtain other private information through session entity authentication. In this protocol, random numbers are added in each round of calculation process, so that the message values before and after change. During round $i+1$ session, the round i session message obtained by eavesdropping is no longer applicable to the current session, and the attacker fails to replay the attack.

Tracing attack: An attacker continuously listens to messages sent by electronic tags, analyzes the location of electronic tags from the messages, and locates the electronic tags. As a result, the electronic tags are damaged and the protocol cannot work properly. However, the protocol messages in this paper are encrypted by introducing random numbers, and the eavesdropping messages of the attacker are all cipher text, which changes every round. As a result, the specific position of the electronic tag is in a constant state of change, and the attacker cannot analyze the specific position of the electronic tag, and the tracking attack fails.

Exhaustive attack: The information obtained by wiretapping, exhaustive way to enumerate private information possible values. In the process of protocol design, not only the hamming weights of encryption parameters are skillfully used, but also at least two or more parameters in each message encryption cannot be known by attackers. When at least two parameters are unknown, the attacker cannot enumerate the possible values of private information by exhaustive method. At the same time, the mixing of random numbers during encryption changes the session messages before and after. When the attacker has not provided the answer to the last session message, the next session has already started, while the last session message has lost effectiveness, making the attacker

spend a lot of manpower and material resources but did not obtain any useful privacy information, and the exhaustive attack fails.

Bidirectional authentication: The reader authenticates the electronic tag T_A through messages A_{11} and A_{12}, and the reader authenticates the electronic tag T_B through messages B_{11} and B_{12}. In this way, the attacker cannot impersonate the electronic tag response reader and ensure the validity of the electronic tag response.

Generation of grouping proof: After the reader first talks with the electronic tag T_A, after the reader talks with the electronic tag T_B, the reader will send the obtained information to the first electronic tag, namely, the electronic tag T_A. Electronic tag T_A calculates A grouping proof message from the information sent by other electronic tags in the group, and then sends the grouping proof message to the reader, that is, A_{13} is the grouping proof message. After the reader receives the GPP, it calculates a grouping proof message based on the received information according to the same algorithm and compares it with the received grouping proof message. Based on the comparison result, it can determine whether the grouping proof is successful. Therefore, the protocol can implement grouping proof.

6 Performance Analysis

The performance of the GPP in this paper is compared with that of other classical GPPs. The detailed analysis results are shown in Table 1.

The symbols in Table 1 above are given the following meanings: symbols indicate that this type of attack can be resisted; Symbol indicates inability to resist this type of attack; The symbol H represents the hash function operation; The symbol ECC stands for ECC operation of ellipse; The symbol P represents the pseudo-random number function operation; The symbol TXO represents traversal XOR operations.

From the perspective of security, the GPP in this paper can resist various common types of attacks, while other GPPs are more or less unable to resist certain types of attacks, which makes the protocol have certain security risks. Therefore, in terms of security, the protocol in this paper is superior to other comparative protocols.

From the perspective of the amount of calculation, especially at the end of the electronic tag is here to calculate the amount, the group proof protocol adopts ultralightweight based on bitwise operations implementation of traverse XOR operations for information encryption, and other documents in the group or encryption algorithm using lightweight or heavyweight encryption algorithm was adopted to realize the information encryption. According to the existing research, the computation amount of one lightweight operation is equivalent to dozens of ultralightweight operations. Therefore, in terms of computation amount, the GPP in this paper has great advantages.

The length of each parameter is the same. The GPP in this paper only needs to store two parameters, so the storage capacity of one end of the electronic tag of the GPP in this paper is 2. Based on Table 1, it can be seen that compared with other protocols, the storage capacity of the protocol in this paper still reduces the storage quantity of parameters.

Tag set number perspective, this paper set proof protocol is given in the fourth Section is suitable for the single tag group now prove that coexistence of detailed steps, then in the eighth Section gives proof protocol scalability of group, suggests that this paper prove that protocol can be used in single tag group coexistence scene, also can use in the tabbed group that coexistence scene, has good scalability. Compared with other protocols, some protocols do not have good scalability.

Based on the above, on the premise of ensuring good security, the protocol in this paper can reduce the overall calculation of electronic tags, and can be applied to the proof coexistence of multiple tag groups, which has good promotion and use value.

7 GNY Logical Formal Proof

This paper adopts GNY logic formalization to conduct formal reasoning analysis of the designed GPP, as follows.

1) Formal model

GNY logic is used to formally analyze the GPP in the paper. According to the convention, R represents reader, T_A represents electronic tag numbered A, and T_B represents electronic tag numbered B. In this paper, the GPP is abstracted in the following form.

$Msg1 : R \rightarrow T_A : ACK$

$Msg2 : T_A \rightarrow R : ID_{A-R}, A_{11}, A_{12}$

$Msg3 : R \rightarrow T_B : ACK$

$Msg4 : T_B \rightarrow R : ID_{B-R}$

$Msg5 : R \rightarrow T_B : R_1, R_2, R_3$

$Msg6 : T_B \rightarrow R : B_{11}, B_{12}$

$Msg7 : R \rightarrow T_A : B_{12}$

$Msg8 : T_A \rightarrow R : A_{13}$

Combined with the calculation of session messages in the previous section, the above abstract results can be summarized as follows:

$Msg1 : T_A \triangleleft *ACK \sim | \rightarrow R| \equiv \#ACK$

$Msg2 : R \triangleleft *ID_{A-R}, A_{11}, A_{12} \sim | \rightarrow T_A| \equiv \#ID_{A-R}, A_{11}, A_{12}$

$Msg3 : T_B \triangleleft *ACK \sim | \rightarrow R| \equiv \#ACK$

$Msg4 : R \triangleleft *ID_{B-R} \sim | \rightarrow T_B| \equiv \#ID_{B-R}$

$Msg5 : T_B \triangleleft *R_1, R_2, R_3 \sim | \rightarrow R| \equiv \#R_1, R_2, R_3$

Table 1: Comparison of security and performance of different group proving protocols

Reference	**Ref[11]**	**Ref[12]**	**Ref[13]**	**Ref[14]**	**Ref[15]**	**Ref[16]**	**This protocol**
Impersonation attack	√	√	×	×	×	×	√
Replay attack	√	√	√	√	√	√	√
Tracing attack	√	×	√	√	√		√
Exhaustive attack	√	√	×	×	√	×	√
Bidirectional authentication	×	√	√	√	√	√	√
Generation of grouping proof	√	√	√	√	√	√	√
Amount of calculation	H	ECC	P	P	H	H,P	TXO
Memory space	3	3	2	3	4	3	2
Number of tag groups	multiple -unit	a set of	multiple -unit	multiple -unit	a set of	multiple -unit	multiple -unit

$Msg6 : R \triangleleft *B_{11}, B_{12} \sim | \rightarrow T_B| \equiv \#B_{11}, B_{12}$

$Msg7 : T_A \triangleleft *B_{12} \sim | \rightarrow R| \equiv \#B_{12}$

$Msg8 : R \triangleleft *A_{13} \sim | \rightarrow T_A| \equiv \#A_{13}$

2) Initialization hypothesis

$A1 : R \ni K_A$

$A2 : R \ni K_B$

$A3 : R \ni ID_A$

$A4 : R \ni ID_B$

$A5 : T_A \ni K_A$

$A6 : T_A \ni ID_A$

$A7 : T_B \ni K_B$

$A8 : T_B \ni ID_B$

$A9 : R| \equiv \#(x)$

$A10 : T_A| \equiv \#(x_A)$

$A11 : T_B| \equiv \#(x_B)$

$A12 : T_A| \equiv T_A \overset{K_A}{\longleftrightarrow} R$

$A13 : T_A| \equiv T_A \overset{ID_A}{\longleftrightarrow} R$

$A14 : R| \equiv R \overset{K_A}{\longleftrightarrow} T_A$

$A15 : R| \equiv R \overset{ID_A}{\longleftrightarrow} T_A$

$A16 : T_B| \equiv T_B \overset{K_B}{\longleftrightarrow} R$

$A17 : T_B| \equiv T_B \overset{ID_B}{\longleftrightarrow} R$

$A18 : R| \equiv R \overset{K_B}{\longleftrightarrow} T_B$

$A19 : R| \equiv R \overset{ID_B}{\longleftrightarrow} T_B$

Initialization assumes that $A1, A2, A3$, and $A4$ are owned by reader R. Initialization assumes that and are owned by electronic tag T_A numbered as A. Initialization assumes that $A7$ and $A8$ are owned by electronic tag T_B numbered as B. Initialization assumes that $A9$ is reader R's belief in the freshness of owning information. The initial assumption is that $A10$ is the electronic tag T_A with the number of A, Initialization assumes that $A11$ is the electronic tag number B, T_B's belief in the freshness of possession information. The initial assumption is that $A12, A13$ are the electronic tag with the number of A, AND the reader believe in sharing information with each other. Initialization assumption is $A14, A15, R$, speaking, reading and writing and the Numbers for A electronic tag T_A believe each other to share information and initialize the assumption, $A16, A17$ is Numbers for B electronic tag T_B trust each other between the read/write device R and share information, initialization suppose $A18, A19$ read/write device is R between electronic tag Numbers for B and T_B trust each other to share information.

3) Prove the objective

Based on the above analysis, there are nine logical formal objectives to be proved in grouping proof coexistence protocol in this paper, which are as follows:

$G1 : R| \equiv T_A| \sim \#(A_{11})$

$G2 : R| \equiv T_A| \sim \#(A_{12})$

$G3 : T_B| \equiv R| \sim \#(R_1)$

$G4 : T_B| \equiv R| \sim \#(R_2)$

$G5 : T_B| \equiv R| \sim \#(R_3)$

$G6 : R| \equiv T_B| \sim \#(B_{11})$

$G7 : R| \equiv T_B| \sim \#(B_{12})$

$G8 : T_A| \equiv R| \sim \#(B_{12})$

$G9 : R| \equiv T_A| \sim \#(A_{13})$

4) Inference proof

In view of the limited space and other factors in this paper, and the proof reasoning process of the nine proof objectives is roughly the same, the proof objective $G1 : R| \equiv T_A| \sim \#(A_{11})$ is only selected as an example for detailed analysis. The reasoning process is as follows:

First, because of the initialization assumption $A10 : T_A| \equiv \#(x_A)$ and the freshness rule

$F1: \frac{P|\equiv\#(x)}{P|\equiv\#(X,Y),P|\equiv\#(F(X))}$, we know: $R| \equiv \#(x_A, ID_{A-L})$.

In $Msg2, T_A \triangleleft * x_A$, that is, $T_A \ni *x_A$, can be obtained by combining initialization hypothesis $A1, A2, A3, A4$ and rule $P2$ simultaneously:$R \ni (x_A, ID_{A-L})$.

Then, from the deduced $R| \equiv \#(x_A, ID_{A-L})$ and $R \ni (x_A, ID_{A-L})$, and according to the freshness rule $F10: \frac{P|\equiv\#(X),P\ni X}{P|\equiv\#(H(X,Y))}$, it can be known:$R| \equiv \#(A_{11})$, that is, $R| \equiv \#(x_A \oplus ID_{A-L})$.

Finally, according to $Msg2$, initialization hypothesis $A15$, deduced $R \ni (x_A, ID_{A-L})$, deduced $R| \equiv \#(A_{11})$, message parsing rule $I3$, we can get: $R| \equiv T_A| \sim (A_{11})$, namely $R| \equiv T_A| \sim (x_A \oplus ID_{A-L})$.

From the definition of freshness, it can be deduced that the proof objective $G1: R| \equiv T_A| \sim (A_{11})$,i.e. $G1: R| \equiv T_A| \sim (x_A \oplus ID_{A-L})$.

8 Scalability of Group Proof Protocol

The GPP given in Section 4 above is the process of proving the coexistence of single tag groups. The above protocol can be extended to prove the coexistence of multiple tag groups as follows:

First of all, reading and writing and the first electronic tags to communicate, to read/write device using the first electronic tags, and after the first electronic tag related privacy information, read/write device will begin to communicate with the second electronic tag, to read/write device using the second electronic tags, and after the second electronic tag related privacy information, Read/write device will begin to communicate with the third electronic tags,$\cdots\cdots$, in accordance with the above way, reading and writing, in turn, communicate with different electronic tags, until the read/write device communicates with the last electronic tag, and the final validation of an electronic tag through reading and writing, speaking, reading and writing device to obtain the final electronic tag related privacy information, The reader will encrypt all the obtained information except the first electronic tag and send it to the first electronic tag. After the first electronic tag receives the message, the first electronic tag contains some privacy information of all the electronic tags in the tag group. The first electronic tag can perform encryption operation, and the result is the grouping proof message, which is sent to the reader. After receiving the tag, the reader can calculate a grouping proof message according to the same algorithm, and then compare the calculated message value with the received message value to realize the coexistence proof of all electronic tags in the tag group. An extensible GPP flow chart is shown in Figure 2.

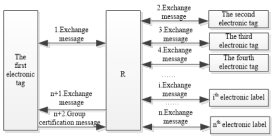


Figure 2: Extensible group proving protocol

9 Conclusion

After analyzing the different defects of the classical GPP, this paper proposes a lightweight tag grouping proof coexistence protocol. The GPP uses traversal XOR operation to encrypt privacy information. Traversal XOR operation can be implemented based on the principle of bitwise operation, which can greatly reduce the total calculation amount of the system. At the same time, traversal XOR operation makes full use of the hamming weight information of encryption parameters, which can reduce the introduction of parameters and reduce the storage cost. The GPP designed in this paper has good expansibility and can be used for coexistence proof of single tag group or multi-tag group. The security and performance analysis of the GPP presented in this paper shows that the protocol can resist common types of attacks such as replay, impersonation, location and tracking, and the total amount of computation is better than other comparison protocols. At the same time, logical formalization is adopted to carry out rigorous reasoning demonstration of the protocol, and the rigorous correctness of the protocol can be deduced.

Acknowledgments

This paper is supported by the 2021 Research on Software-defined Key Technology of Industrial Control Real-time Network (China) (2021FNA0210); Guangdong 2021 Project of Educational Science Planning (Special program of Higher Education)(2021GXJK438).

References

[1] Y. P. Duan, "Lightweight RFID group tag generation protocol," *Control Engineering of China*, vol. 27, no. 4, pp. 751–757, 2020.

[2] F. Zhu , P. Li , H. Xu , et al, "A lightweight RFID mutual authentication protocol with PUF," *Sensors*, vol. 19, no. 13, pp. 2957–2978, 2019.

[3] K. Fan , W. Jiang , H. Li , et al, "Lightweight RFID protocol for medical privacy protection in IoT," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 4, pp. 1656–1665, 2018.

[4] W. Liang , S. Xie , J. Long , et al, "A double PUF-based RFID identity authentication protocol in service-centric internet of things environments," *Information Sciences*, vol. 50, no. 3, pp. 129–147, 2019.

[5] H. Y. Kang, "Analysis and improvement of ECC-based grouping-proof protocol for RFID," *International Journal of Control and Automation*, vol. 9, no. 7, pp. 343–352, 2016.

[6] P. Wang , Z. P. Zhou , J. Li, "Improved serverless RFID security authentication protocol," *Journal of Frontiers of Computer Science and Technology*, vol. 12, no. 7, pp. 1117–1125, 2018.

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

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

[9] Z. B. Zhou , P. Liu, "An anonymous offline RFID grouping-proof protocol," *Future Internet*, vol. 10, no. 2, pp. 1–15, 2018.

[10] S. Y. Chiou , W. T. Ko , E. H. Lu, "A secure ECC-based mobile RFID mutual authentication protocol and its application," *International Journal of Network Security*, vol. 20, no. 2, pp. 396–402, 2018.

[11] Z. J. Cao , O. Markowitch, "Analysis of shim's attacks against some certificateless signature schemes," *International Journal of Network Security*, vol. 23, no. 3, pp. 545–548, 2021.

[12] X. R. Deng S. Q. Mei, "Mobile RFID bidirectional authentication protocol based on shared private key and bitwise operation," *Computer Applications and Software*, vol. 37, no. 7, pp. 302–308, 2020.

[13] Z. H. Liu , C. J. Huang , H. Suo, "Modifified mobile RFID bidirectional authentication protocol against counterfeiting attack," *Computer Applications and Software*, vol. 37, no. 6, pp. 309–315, 2020.

[14] F. TAN, "An improved RFID mutual authentication security hardening protocol," *Control Engineering of China*, vol. 26, no. 4, pp. 783–789, 2019.

[15] X. H. Zhao, "Attack-defense game model: Research on dynamic defense mechanism of network security," *International Journal of Network Security*, vol. 22, no. 6, pp. 1037–1042, 2020.

[16] J. F. Chong , Z. P. Zhuo, "Constructions of balanced quaternary sequences of even length," *International Journal of Network Security*, vol. 22, no. 6, pp. 911–915, 2020.

Biography

Yu-zhen Li received a master's degree in School of Computers from Guangdong University of Technology (China) in June 2016. Her current research interest fields include information security.

Wen-tao Zuo received a master's degree in School of Computers from South China Agricultural University (China) in June 2010. He is now a lecturer, working in Guangzhou College of Technology and Business. His current research interest fields include information security.

Dao-wei Liu received a master's degree in School of Computers from Guangdong University of Technology (China) in June 2016. His current research interest fields include information security.

A New Scheme of BACnet Protocol Based on HCPN Security Evaluation Method

Tao Feng, Xiao-yan Jiang, Jun-li Fang, and Xiang Gong
(Corresponding author: Xiao-yan Jiang)
School of Computer and Communication & Lanzhou University of Technology
Lanzhou,Gansu 730000,China
Email: 1848469146@qq.com
(Received Jan. 17, 2020; Revised and Accepted Oct. 9, 2022; First Online Oct. 15, 2022)

Abstract

They are building automation or management systems control services such as heating, air conditioning, and safe aisles in a facility. The standard protocol used to transmit data about the status of components is BACnet. To solve the security problem of BACnet protocol device authentication of intelligent building communication protocol [1], the attack vectors and security requirements of BACnet protocol device authentication are analyzed. First, this paper verifies the consistency of BACnet protocol device authentication based on Petri net theory and CPN Tools [14]. It introduces an improved Delov-Yao attack model to evaluate the protocol model security to verify whether there are other undiscovered attacks. Secondly, a new BACnet protocol device authentication model is proposed to solve the problems of session key leakage and message tampering during device authentication. This mechanism needs to change the key distribution method and introduce random numbers to complete BACnet protocol devices. Certified. The new scheme can use BACnet protocol devices to authenticate cryptographic primitives without significantly upgrading existing platforms. CPN Tools have verified the protocol; the results show that no intrusion path can ensure device certification's integrity, authenticity, freshness, and confidentiality.

Keywords: BACnet Protocol; CPN Tools; Formal Analysis

1 Introduction

With the rapid development of information technology, intelligent building to become "Internet +" and the construction industry and the direction of the depth of integration of a breakthrough, but the explosive growth of network security vulnerabilities, a large number of mobile Internet application of new technologies, automatic control, gave the introduction of intelligent building new information security risks. A growing number of cyber attacks show that the intelligent building is unsafe [2,13,19]. TCP / IP protocol data communications technology is widely used in intelligent building systems based, has been achieved despite the requirements for intelligent remote monitoring of construction equipment, but the original data communication protocol network face greater threat of attack.

The widespread use of the BACnet protocol in the field of intelligent building systems has proved to be unsafe [3, 5, 16]. Because of the BACnet protocol's Internet connectivity and the ability to find BACnet devices using the SHODAN search engine (cf.www.shodanhq.com), BACnet devices can also be remotely attacked, for example, by smoke detectors or other important BAS devices. Therefore, BACnet protocol must be studied and improved from the perspective of equipment authentication of both sides of communication to ensure the security of communication. The BACnet protocol standard defines network security services, which provide security mechanisms for communication equipment identification, data source identification, operator identification, and data confidentiality and integrity. However, few building automation system suppliers have implemented it. An attacker may exploit this vulnerability to intercept the session key and tamper with the message to modify the command of the communication device or perform an unauthorized service to attack remotely.

Literature [17] proposes a way to improve the reliability and security of networks and applications using traffic standardisers, but the tool does not implement prevention techniques. Literature [7] [10] made a detailed study on the identification problem, denial of service, eavesdropping and buffer overflow in the core functions of the protocol, and proposed a deterministic improvement on the BAS networking problem that was not taken into account at the beginning, and added the remote management technology of enterprise internal network and Internet connection. Literature [6] mainly discusses the limitations of secure communication and the security of data exchange in BAS. Holmberg *et al.* proposed corresponding

mitigation measures for some of the identified vulnerabilities in BACnet, such as BACnet firewall [23], which, however, required dedicated hardware due to its high computational complexity. Literature [8] determined the ability of legitimate malicious commands running within BACnet works to prevent them from transmitting data traffic through boundary firewalls. Literature [11, 21] focused on this problem and proposed a potential solution for BAS specific intrusion detection systems (IDS). Above the BACnet protocol security research mainly focus on its function and connect to the Internet after a series of problems, for internal data transmission security agreement did not put forward effective safety assessment methods, and puts forward improvement scheme couldn't resist the attacker as communications equipment for the session key and tampering with BACnet server and client attacking threat. Therefore, based on literature [9], guided by colored Petri net theory and DelovYao attack method, and based on CPN Tools model detection tool, this paper focused on the formal modeling and security assessment of the protocol, explored protocol vulnerabilities, proposed targeted security improvement schemes, and applied CPN detection Tools to verify the security of the proposed schemes.

This paper is organized as follows. In Section 3, based on Petri net theory and CPN Tools tool, we verify the consistency of the BACnet protocol device authentication. In Section 4, the improved Delov-Yao attack model of safety assessment protocol model is introduced, we verify that there are other attacks undiscovered. In Section 5, we propose a new protocol BACnet device authentication model, which needs to change the key distribution method. Meanwhile the random number is introduced to complete the BACnet protocol device authentication, then we use the CPN Tools tool to verify the security of the scheme and analyze its performance. Finally, In Section 6, we describe the general conclusions.

2 Preliminaries

2.1 BACnet Protocol Overview

BACnet is an object-oriented peer-to-peer network protocol, in order to ensure the efficiency of communication,we use the OSI-RM streamlined architecture model to define application layer, network layer, data link layer and physical layer. BACnet focus on the network layer and above, the goal of which is to run on any data link and physical media. The BACnet standard defines the data structure that represents the communications of devices on BACnet. The core data structure is an object with 54 standard types, as shown in the BACnet architecture hierarchy diagram in Figure 1.

BACnet standard defines six functional categories of services [12]: Object Access Service, file access service, alarm and event services, remote device management services, virtual terminal services and network security services. These services include all aspects of the building's

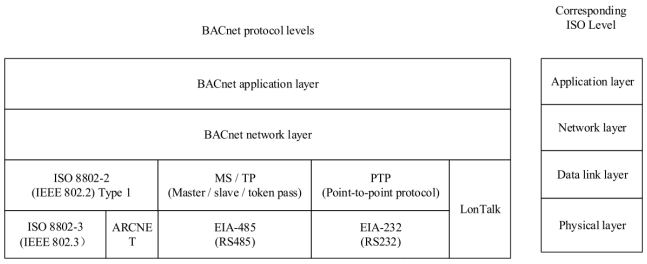


Figure 1: BACnet architecture hierarchical graph

automatic control network system. With the development of the protocol and the application of the network, new services will be added.

2.2 Network Security Services

In the BACnet standard, network security services is an optional content [18]. BACnet network security mainly provides security mechanisms for equipment authentication, data source authentication, operator identity authentication, and data confidentiality and integrity [4], but does not provide communication security mechanisms such as access control and non-repudiation. This paper focuses on the analysis of equipment certification..

In the BACnet standard, there is a flaw in device authentication security, which is vulnerable to attacks such as man-in-the-middle ,replay attacks and other methods. For example, in the BACnet standard security service, BACnet's key distribution is a local method. Its keys are stored in the local device, there is no method for defining key updates, an attacker is able to crack an old session key and continue the session with that key. In network security services, key exchange, generation, distribution, storage, and erasure are critical to the security of the key. The BACnet standard has no systematic definition for this.

2.3 CPN Tools Modeling Tools

CPN [20] is a graphical language that plays an important role in modeling and verifying concurrency, distributed systems and other systems.CPN is a discrete behavioral model language combined with the capabilities of a high-level programming language, which provides basic graphical representation and the ability to model concurrent, communication, and synchronous. CPN's ML (Markup Language) [22] programming Language is based on the standard ML programming language. It provides basic data type definitions (complex data types can be combined through products, unions, etc.), a description of data manipulation, meanwhile, it can create a compact and parameterized model. CPN's model language is a general-purpose modeling language. It is not only applicable to a class of systems, but is oriented to a wide range of systems and can describe concurrent systems.

Its typical application fields include communication protocols, data networks, distributed algorithms, embedded systems, and many applications in the industrial field.

CPN Tools is a computer-aided design tool developed by Danish researchers for protocol modeling, analysis and validation. CPN tools can be used for CPN model editing, simulation, state space analysis, and performance analysis. CPN tools support tools the time and timeless level CPN models. CPN tools is a computer tool for industrialization. It can use simulation functions to investigate the behavior of model systems, and use state space methods and model detection methods to verify attributes. The interaction between the user and the CPN tools is based on the interactive technology that directly manipulates the graphical representation of the CPN model. Its representation is intuitive and has many industrial applications.

3 BACnet Protocol Modeling Device Authentication HCPN

3.1 BACnet Protocol Device Authentication Message Flow Model

Authentication Message Flow (MSC) model is shown in Figure 2. ReqKey represents a key request to the server, Ks represents the session key distributed by the server to devices A and B, IDa indicates the identity of device A, Kb denotes the master key of device B, IDb represents the identity of the device B, Ka represents device A master key. Authenticate represents peer entity requests the service identification, Pseudo Random Number represents a pseudo-random number in the packet, ComplexACK indicates a complex response message Modified Random Number response message indicates the modified random numbers.

Authentication mode as follows:

1) Run the initialization algorithm, devices A, B uses the DES algorithm to generate its own master key. Device A has the master key Ka and device B has the master key Kb (Shared only with the key Server);

2) A sends a "ReqKey" request to the server, requesting to obtain the session key Ks;

3) After receiving the request message from A, the key server Server generates a session key Ks, encrypts Ks and IDa using Kb of device B, and sends it to device B;

4) Device B uses Kb to encrypt its identity IDb and send it to the key server Server. The key server Server verifies the address of device B;

5) The key server Server then uses the Ka of the device A to encrypt Ks and IDb and sends it to the device A; server Server. The key server Server verifies the address of device B;

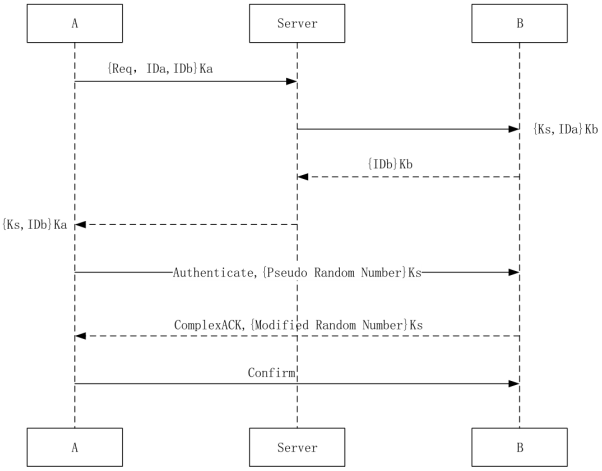


Figure 2: Authentication mode message flow model (MSC)

6) Device A receives Ks and begins to identify device B, device A generates an authenticate request, the protocol data portion of which and Pseudo Random Number are encrypted with Ks and sent to B;

7) Device B decodes the authentication request from A, modifies the Pseudo Random Number to Modified Random Number, and uses Ks to encrypt and return a ComplexACK message to device A;

8) Device A decrypts the message after receiving it, and if a ComplexACK message containing the correct Modified Random Number is received, device authentication succeeds.

3.2 BACnet Protocol Defined Color Set Device Authentication Message

First, analyze the four messages that the key is distributed. The information elements needed to establish the model include the identity of device A and device B, the master keys (Ka and Kb) of the two devices themselves, and the session key Ks of the key distributor; There are two formats for encryption and decryption. One is device A and device B use the master key Ka and Kb to encrypt their identity and request information, The other is that the key distributor uses the keys Ka and Kb of the device A and device B that are known in advance to decrypt the obtained information to verify the identity; We are combining basic information elements and cipher text into four message formats. Next, analyze the device authentication message, both messages are also ciphertext, and the content is differentand, Therefore, the task of cipher text is omitted when setting the color and the message is directly defined on the basis of the information elements. Finally, the main color collection as defined in Table 1.

Table 1: BACnet protocol devices certified color set statement

Category	Key element	Color set definition
Key distribution	MSG1	colset MSG1=product ID*CRY2
	MSG2	colset MSG2=product ID*CRY1
	MSG3	colset MSG3=CRY2
	MSG4	colset MSG4=MSG2
Equipment certification	ASK	colset ASK=record m:MSG*k:KEY
	RSP	colset MSG4=colset RSP=product NONCE*NONCE2
	RPL	colset RPL=record r:RSP*k:KEY
	CFM	colset MSG4=colset CFM=record n:NONCE*k:KEY2
Data	PACKET	colset MSG4=colset PACKET=union MSG1+MSG2+MSG3+MSG42

Here,ID represents the device, KEY represents the key that appears during the authentication process, and NONCE represents the pseudo random number Pseudo Random Number and the modified random number Modified Random Number. The data packet type (colset packet) is uniformly organized using the union type, and its elements are specific descriptions of different types of data packets. Among them, the MSG1 type is used to describe the message of Step 1 of the protocol operation of the data exchange between the session initiator and the key distributor; the MSG2 type is used to describe the message of Step 2 between the key distributor and the session responder; The MSG3 type is used to describe a message with a step of 3 between the session responder and the key distributor; the MSG4 type is used to describe a message with a step of 4 between the key distributor and the session initiator.The ASK record type is used to describe the protocol request message sent by the session initiator to the responder when the protocol runs in Step 5. The RPL record type is used to describe the response data message sent by the session responder to the initiator when the protocol runs in Step 6. The CFM record type is used to describe the data message that the session initiator confirms the received information when the protocol runs in Step 7.

3.3 BACnet Protocol Device Authentication Model HCPN

This section will be established protocol BACnet device authentication HCPN hierarchical model, through the use of alternative high-level transitions in the network CPN tools tool to link it to a more detailed sub pages to hide the details of the top layer of the model , and a high-level description of a simplified model presented can be used broadly defined from the picture system. The hierarchical BACnet protocol model includes top-level and sub-pages of each entity-level model. Top model is an abstract description of the overall agreement, Entity layer model provides implementation details of the HCPN model.

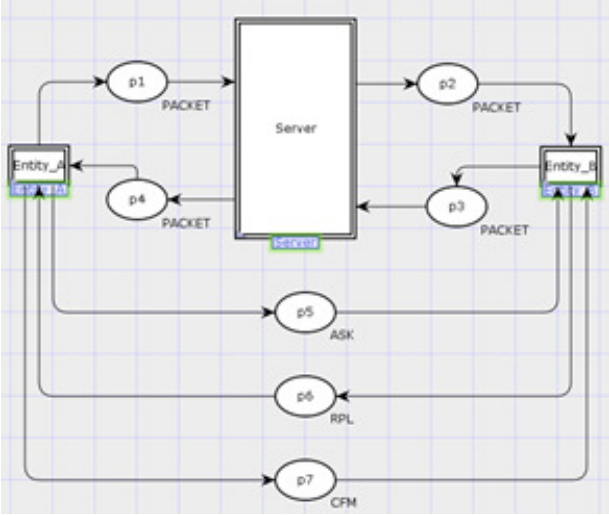


Figure 3: BACnet protocol device authentication CPN top level model

3.3.1 BACnet Protocol Device Certified Top Model

The CPN top-level model of the BACnet protocol device certification is an abstract description of the entire communication protocol, including the communicating parties, the communication network, and the messages passed. As shown in Figure 3, the double-line rectangle in the figure is an alternative change, and the oval is the message places.The left-hand alternative transition Entity_A represents the communication device A, the middle alternative transition Server represents the key distributor, and the right-most alternative transition Entity_B represents the communication device B.The top-level model completely simulates the BACnet protocol device authentication session process, including the key distribution process and device authentication process, and the processing of key information, which is a high degree of abstraction and generalization of the protocol MSC model.

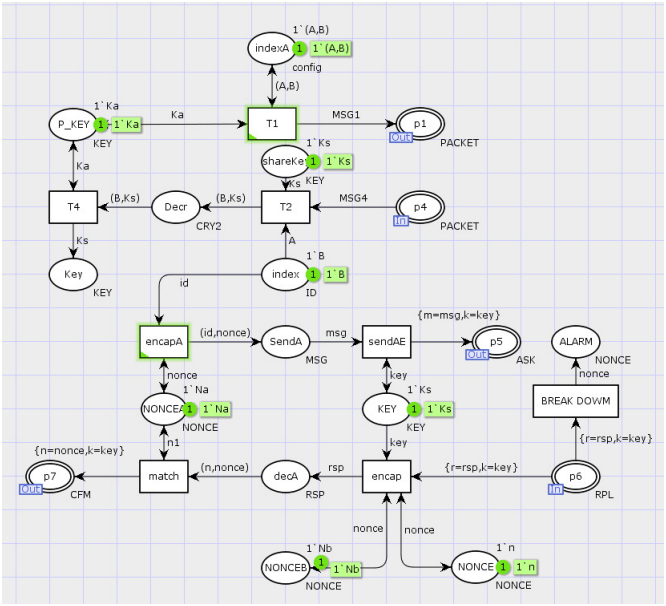


Figure 4: BACnet protocol CPN model device authentication entity A

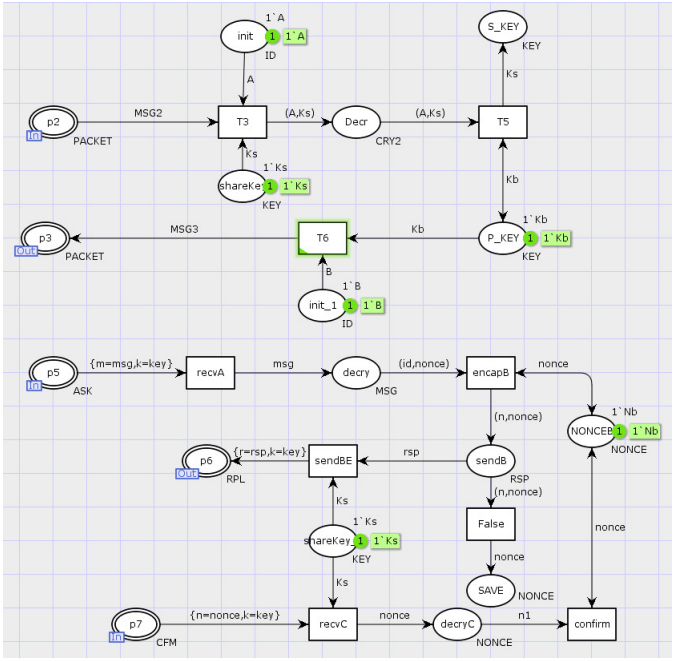


Figure 5: BACnet CPN model entity device authentication protocol B

3.3.2 BACnet Protocol Physical Layer Device Authentication Model

As shown in Figure 4, the model for entity A in the BACnet protocol device authentication for the protocol includes 16 message places and 7 transitions. The process of sending and receiving data packets where entity A requests a session from the key distributor and initiates identity authentication to entity B is described. In the model, the fusion place index is used to configure the session participation mode, and the protocol initiates the session and the role and identity information of each entity; For entity A, when participating in the process of session initiation (protocol execution Step 1), the session participation mode configuration controls the initiator and responder of the data it sends. Because entity A is an honest entity, its identity is A , And the respondent is entity B in this model, generate the shared keys Ka and Kb with the key distributor, and save them to the corresponding place P_KEY, organize and send MSG1 type data to the communication channel port place p1; the participating protocol executes Step 4. The key distributor sends data of type MSG4 to the communication channel port place p4 while receiving the data of entity B. the initiator uses the shared key Ka with the key distributor to decrypt to obtain the session key Ks. Fusion place index order of a session configuration settings, such changes in the entity performing the ignition operation of the respective step protocol.

As shown in Figure 5, the model of entity B in the BACnet protocol device authentication of the protocol includes 16 message places and 8 transitions. It describes the entity B receives the key distributor session key distribution, and data packets received authentication entity A initiates a transmission and reception process. Fusion places used in the model for the session participation mode configuration index, set the protocol to initiate the session, and where the role of each entity's identity information and the like taken; For entity B, the session participation Step 2, Step 3 and Step 6. In MSG2, entity B decrypts the received data with the shared key Kb with the key distributor and obtains the session key Ks and the identity of the initiator entity A, and saves the obtained session key in a specific repository. The identity of the corresponding entity in S_KEY is stored in the place init. It will then send MSG3 type and key distributor for authentication. Fusion place index order of a session configuration settings, such changes in the entity performing the ignition operation of the respective step protocol.

As shown in Figure 6, the model of the key distributor in the BACnet protocol device authentication of the protocol includes 10 message places and 3 transitions. This model describes the process by which the key distributor Server distributes the session keys for identity authentication for entities A and B. Step 1: Receive the message MSG1, determine whether the encryption key is Ka, use Ka to decrypt it, obtain the identity of the session initiator and the identity of the responder, and save it to the corresponding place init, Step 2: according to the identity of the responder in the message MSG1, organize and send MSG2, and send the session key Ks and the identity of the session initiator to the responder; Step 3: the key distributor receives and uses the shared key Kb with the key distributor to decrypt the message MSG3 sent by the responder, and determines whether the identity of the responder is correct. Step 4: according to the

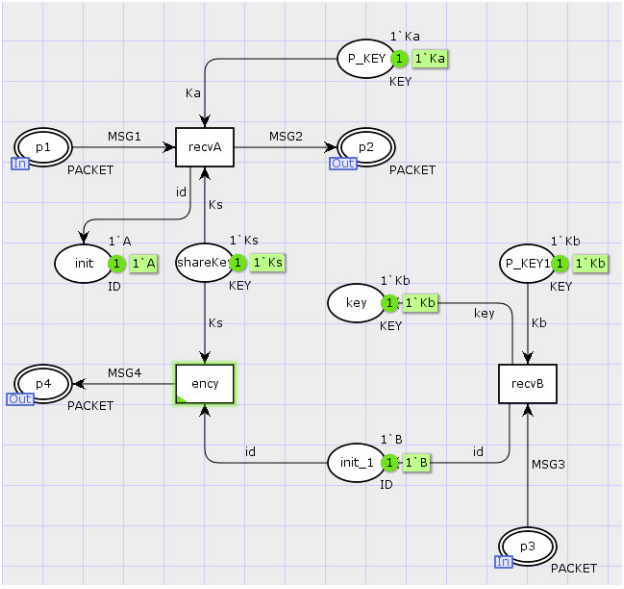


Figure 6: BACnet protocol CPN model device authentication key dispenser

identity of the initiator in the message MSG1, use Ka to encrypt the session key Ks and the identity of the responder, and organize and send the message MSG4.

3.4 Features the Original Model of Consistency Verification

Whether the original model accurately reflects the functionality required by protocol specification to determine the effectiveness of subsequent security assessment model. In this section, a state space analysis tool will be used to verify the functional consistency of the HCPN model of the original BACnet protocol device certification described above. During the verification process, it is mainly verified whether the behavior of the key distribution process and the device authentication process conforms to the authentication behavior attributes described in the protocol specification. It should be noted that the original HCPN model did not introduce network attacks.

3.4.1 Analysis of Results Expected

For any CPN model state analysis, researchers mainly investigate the activity of their state nodes, state master nodes, andtransitions, so as to compare with the expected system state, determine whether the model is consistent with expectations, and meet the protocol's behavior specifications. As shown in figures 4 and 5, according to the BACnet protocol device authentication specifications, when a device initiates an authentication request, it will also generate a pseudo-random number of the desired data packet. Whether the device can successfully authenticate depends on the encrypted data verification, When authentication is successful, device A will trigger the transition encap and match without triggering the transition BREAK DOWN, and device B will trigger the transition recvC and confirm without triggering the transition False, so the transition BREAK DOWN and False can be predicted in the model Two dead transitions. In addition, the original model will finish running after the authentication request is completed, so it can be predicted that there is no live transition of the model termination state and there is only one dead state node. Table 2 gives the expected performance results of the original model.

3.4.2 Analysis of the Results of the State Space

This section is mainly used for state space analysis tool for behavioral attributes protocol model for analysis. SML includes the following query: SccReachable (). Nodes exist in the model for determining the size, and determining the possibility of deadlock in the model; InitialHomeMarking (). Means for determining whether there is always reachable from any other accessible STATUS This state is a state of the main model; ListDeadMarkings (). Model used to determine the final state; ListDeadTIs (). Used to determine whether a given change is not up to the state to enable; TisLive (). For determining whether changes always occurring or being performed; Reachable (x, y) for determining whether the marker M (y) up to the path from the marker M (x). Table 3 shows the state space model of the original agreement query results.

4 BACnet Protocol Device Authentication Security Assessment

4.1 Based on BACnet Protocol Attacker Model of Device Authentication Security Assessment

This paper makes full use of the advantages of CPN visual modeling, dynamic model execution, and state space analysis of system operation. Based on the improved solution of the Delov-Yao [15] attacker model, the attacker is introduced into the HCPN model of BACnet protocol device certification. The model evaluates the security of the protocol and analyzes the loopholes in the protocol.

Figure 7 shows the security assessment model of the Server subpage of the HCPN model server authentication based on the BACnet protocol. The server subpage simulates the key distributor to distribute the session keys for devices A and B. According to the assumption of the Delov-Yao attack, the attacker has eavesdropping, tampering, and packet loss, and can disguise as the session initiator and responder, but not as the trust third-party servers. As shown in the figure, the transitions in the red annotation and the place simulate the replay attack. The transition t0 intercepts the message sent by the protocol in the first step, defines the color set of the place

Table 2: expected performance of the original model results

Types	Death changes	Live changes	The master node	Dead state
Numbers	2	0	0	1
Name	BREAK DOWN,False	/	/	/

Table 3: expected performance of the original model BACnet protocol device authentication HCPN state space model results

Types	Numbers	Name
State space node	5806	/
State Space arcs	15767	/
Scc Graph node	5806	/
Scc Graph arcs	15767	/
Home Markings	0	/
Live Transition	0	/
Dead Markings	1	/
Dead Transition	2	BREAK DOWN,False

resovle as DB, stores the decomposition and messages to be decomposed; defines the place The color set of P5‘ is CB, which stores synthesized and band-synthesized messages; defines the color set of element in the place as AB, and stores atomic messages. Transition t11 uses the rules of the attacker to save the messages that cannot be decrypted to the place P5‘ using excessive rules. Transition t3: uses the attacker's composition rule to synthesize the atomic message and save it to the place P5‘. The concurrency control place SP is used to limit the transition t3 corresponding to the composition rule. The transition t4 sends the attacker's synthesis message to the channel p2 port place. Purple section marked transition guard simulated spoofing attacks, including transitions recvA, ency, recvB. The arc expression labeled in the blue part of the transition At simulates a tampering attack.

4.2 BACnet Protocol Device Authentication Security Property Validation Analysis

When the model is successful and there is no semantic errors, you can run "into the state space" tool. Then run the "Calculate State Space" tool and the "Calculate Strongly Connected Component Diagram" tool in order. If the operation is successful, it means that a state space for the protocol model has been generated. You can now save a Standard State Space Report file. The model status query results are shown in Table 4.

Security evaluation model with the introduction of improved Delov-Yao attacker model lead to a substantial increase compared to the original model state space of nodes and the number of arcs, is in line with expecta-

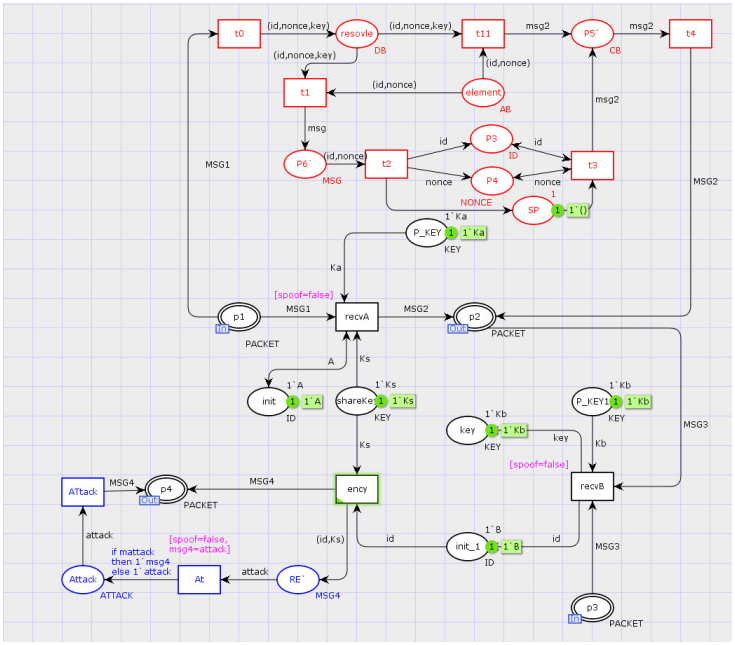


Figure 7: BACnet protocol device authentication model attacker

Table 4: BACnet protocol device authentication model introduced HCPN attacker's state-space search results

Types	Numbers
State space node	18751
State Space arcs	50922
Scc Graph node	18751
Scc Graph arcs	50922
Home Markings	27

tions, and the number of state space arc, the number of nodes and the number of strong connectivity arc, the same number of nodes, indicating safety assessment model for all state nodes are reachable lead to the occurrence of the state of infinite loop iterations and behavior do not exist, which further illustrate the improved Delov-Yao attacker model is valid.

As can be seen from the report, the CPC model generates a 18,751 node state space where the dead node 27, the results show that the addition of three kinds under attack mode security assessment occurred unpredictable behavior. Use ListDeadMarking () to determine the number of 27 dead nodes. After checking the status of all dead nodes, it was found that the messages received by the key distributor Server in nodes 164 and 184 were sent by the intruder, and then the intruder tricked the key distributor Server to obtain the session key and tampered with the correct message Send false messages; it is found in node 549 and node 572 that the message received by responder B has been changed by the intruder, and the intruder can send the same message to responder B repeatedly.

5 BACnet Device Authentication Protocol of The New Program

According to the security assessment of the above method, it can be known that the security result of the BACnet protocol device authentication actually does not meet the authentication requirements in the specification, and it cannot resist two attacks of replay and tampering. In view of the above security threats, this article proposes a method that introduces the generation of random numbers and changes the distribution method of session keys, including the security improvements in the key distribution phase and the device authentication phase, and using HCPN again The model verifies the performance and security of the improved protocol.

5.1 BACnet Protocol Device Authentication Security Scheme Based on New Modeling HCPN

The results of the security assessment of the BACnet protocol reflect that the protocol does not actually meet the authentication requirements claimed in the BACnet standard and cannot withstand replay and tampering attacks. Aiming at the above security threats, this section introduces methods for adding random numbers and changing the session key distribution process. This paper proposes an improved protocol and verifies the improved performance security.

The improved authentication message flow (MSC) model is shown in Figure 8. ReqKey indicates requesting a key from the server, Ks indicates the session key distributed by the server to devices A and B, IDa indicates the identity of device A, Kb indicates the master key of device B, IDb indicates the identity of device B, and Ka indicates the identity of device A Master key. Authenticate indicates that the peer entity requests the service, Pseudo Random Number indicates the pseudo random number in the message, ComplexACK indicates the complex response message, Modified Random Number indicates the modified random number of the response message, and Na and Nb are random numbers added by the improved protocol .

Authentication modes for improved as follows:

1) Run the initialization algorithm. Devices A and B use the DES algorithm to generate their own master keys. Device A has a master key Ka, and device B has a master key Kb(shared only with the key server Server);

2) Device A sends a message(ReqKey = IDa,IDb,Na)to the key server Server, requesting a session key to secure the logical connection to device B. This message contains the random number Na and the device A for this transmission. And the identity of device B;

3) After receiving the message from device A, the key server Server uses Ka to perform the data source identification process, determine whether the request is issued by device A, and then use the DES algorithm to generate the session key Ks. The key server Server uses the master key Ka of the device A to encrypt the one-time session key Ks for the session and the previous request information, and uses the master key Kb of the device B to encrypt the one-time session key Ks and The identity of device A is encrypted and sent to device A, which is (E (Ka,[Ks,IDa,IDb,Na])E(Kb,[Ks,IDa]));

4) After receiving the message sent by the key server Server, device A uses Ka to perform the data source identification process to determine whether it is sent by the key server Server. Decrypt and store the session key Ks to be used, and send the information from the key server Server to device B, which is E(Kb,[Ks,IDa]);

5) After receiving the message, device B decrypts and obtains the session key Ks and the device A (IDa) of the other party who wants to establish a connection. Device B uses the new session key Ks to encrypt the random number Nb of the transmission and sets the result. Send to device A, namely E,(Ks,Nb);

6) After receiving the message from device B, device A starts to authenticate device B. Device A uses the session key Ks to encrypt the Authenticate service request, that is, it contains the data part of the request protocol, Pseudo Random Number, and random number Nb, and sends it to device B;

7) Device B decodes after receiving the authentication service from device A. First verify whether the received random number Nb is the same as the previous one. If it is the same, then reverse the highest

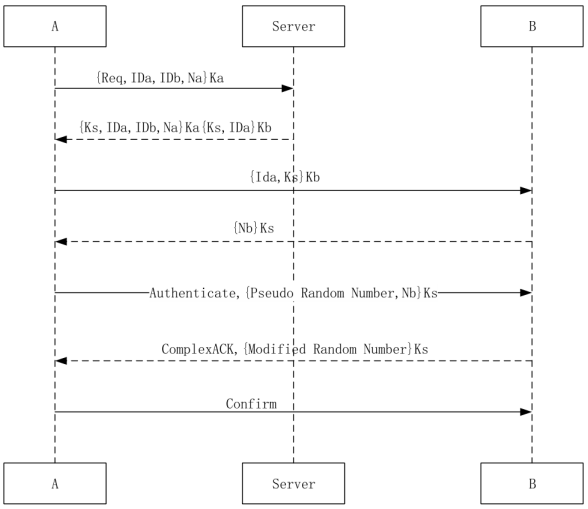


Figure 8: Improved authentication mode message flow model (MSC)

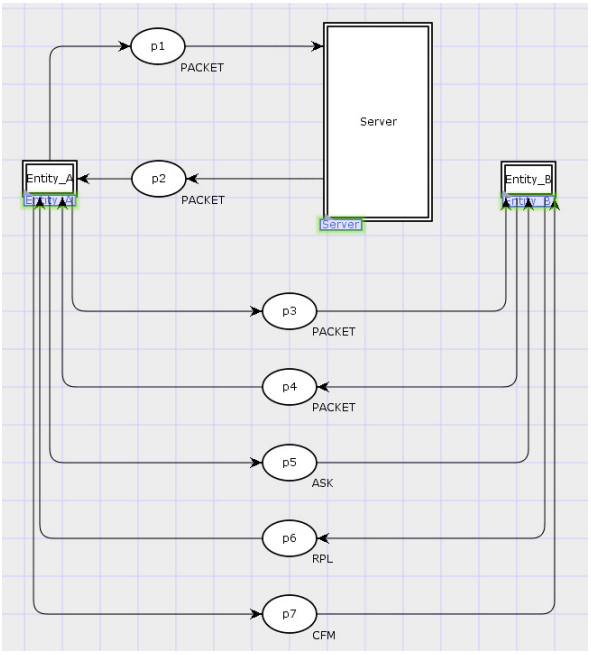


Figure 9: the improved device authentication BACnet protocol CPN top level model

and lowest bits of each byte of this parameter, modify the Pseudo Random Number to Modified Random Number, and return a ComplexACK (Transfer service request has been successfully executed) message;

8) Device A decodes the received response message to check whether the ComplexACK message contains the correct "Modified Random Number". If it is correct, then device B completes the identity authentication.

5.2 The New BACnet Protocol Device Authentication Model HCPN

This section will establish the improved HCPN hierarchical model for BACnet protocol device authentication, which mainly includes the top-level model, device A model, device B model, and key distributor Serves model.

1) Equipment Certification of improved CPN top model

Figure 9 shows the improved top-level model of the CPN. The model includes both parties to the communication, the communication network, and the passed messages. It consists of 3 alternative transitions and 7 places. It completely simulates the complete communication process of key distribution and device authentication. The left-hand alternative transition Entity_A represents the communication device A, the middle alternative transition Server represents the key distributor, and the right-most alternative transition Entity_B represents the communication device B.

2) CPN model entity A device authentication improved

The behavior of entity a mainly includes sending session key request message to key distributor servers, including random number Na and device ID; for entity B, entity a first has the function of distributing key, after authenticating that both sides of the device have session key KS, entity a sends an identity verification request to B, and verifies the correctness of identity by verifying protocol data part and complex message complexack Sex. Figure 10 shows the CPN model of entity a of the improved device authentication.

3) CPN model of device authentication entity B improved

Figure 11 shows the CPN model of the improved device authentication entity B, which includes 7 transitions, 4 places ports and 11 general places. Its behavior includes receiving the session key KS sent by device a, sending the random number Na for authentication to device a, and finally receiving the data authentication of the device authentication request. Among them, the functions of transition recvb, encapa and the model before improvement are similar.

4) Improved device authentication key distributor of CPN model Servers

The behavior of key distributor services mainly includes decrypting the message from device a, obtaining the identity ID and random number Na of device a and B, and then encrypting the session key KS in two messages to device a. Figure 12 shows the CPN model of the improved BACnet device authentication key distributor services, which includes three transitions, two places ports and five common places.

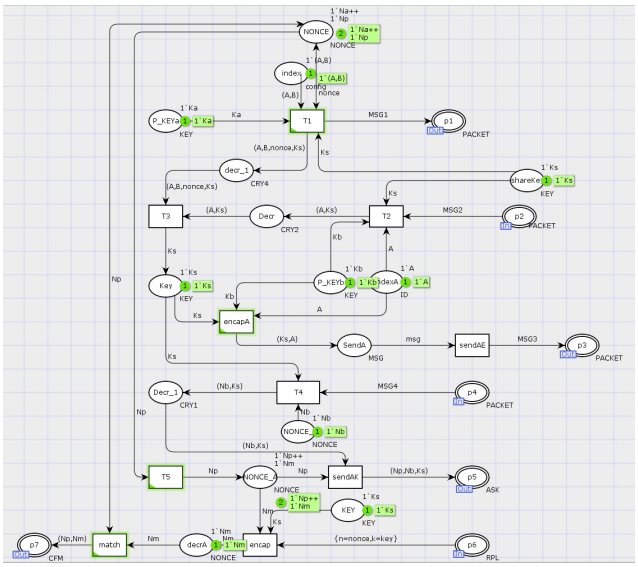


Figure 10: CPN model BACnet protocol entity A device authentication improved

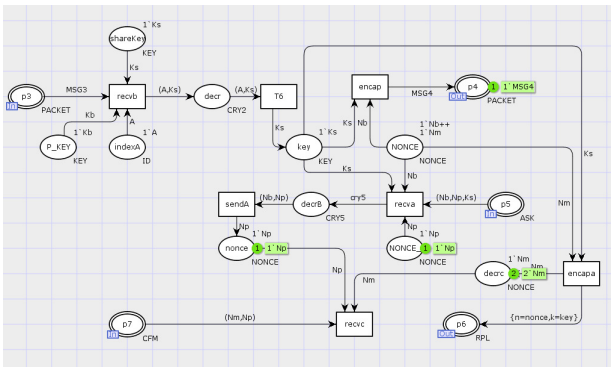


Figure 11: CPN model entity B improved BACnet protocol authentication device

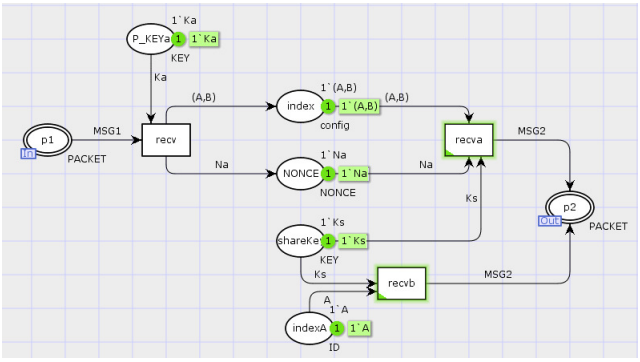


Figure 12: BACnet protocol device authentication key distribution is improved Serves the CPN model

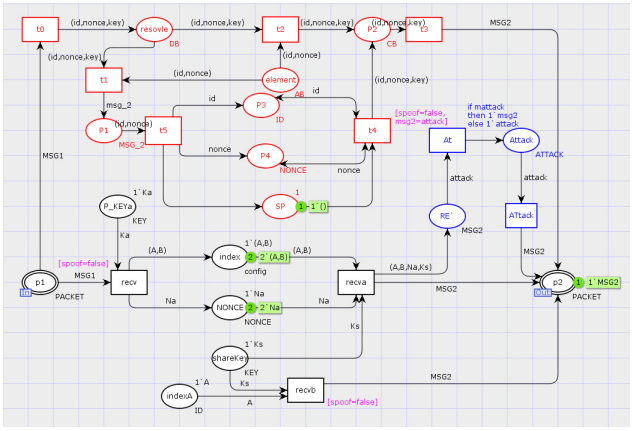


Figure 13: Equipment Certification of improved security evaluation model CPN

5) New equipment certified CPN security evaluation model

Figure 13 shows the security evaluation model of the improved BACnet protocol device authentication HCPN model server sub page. Server sub page simulation key distributor mainly distributes session key for device a, and the improved servers will no longer have data interaction with device B. According to the Delov-Yao attack hypothesis, the attacker has eavesdropping, tampering, packet loss, and can disguise as a session initiator and responder, but not as a trusted third-party server. As shown in the figure, the transitions of the red part and the places simulate replay attack, the transitions of the purple part simulate deception attack, including transitions recv, recvb, recva, and the blue part simulate tamper attack, including transitions At and ATtack.

5.3 The New BACnet Protocol Device Authentication Security Assessment

Table 5 shows the comparison of the state results of the BACnet protocol device authentication security evaluation model after the improvement and before the improvement. Due to the introduction of random Numbers and the change of the key distributor and the way of distributing keys, the model increases the number of transitions and places. Compared with that before the improvement, the number of states and arcs is significantly increased.

In the safety assessment phase, add parameters to verify the BACnet protocol attacks Equipment Certification improved tampering and replay can withstand two attacks. Table with the improved die before improvement number of nodes is reduced from 42 to 5, the above statement SML attacks investigations have found that all dead attack state causes nodes, reducing the number of dead nodes showed increased change key distribution random number and after manner, an attacker can not get the

Table 5: Comparative state space model before and after authentication security assessment improved apparatus BACnet protocol

Types	Improved ago	The improved
State space node	18751	112506
State Space arcs	50922	305532
Scc Graph node	18751	112506
Scc Graph arcs	50922	305532
Home Markings	27	5

full details of the message, including the session key, devices a, B of the master key and the random number, the device authentication BACnet protocol improved to overcome information message tampering and replay attacks, to meet the BACnet protocol apparatus certification certification requirements specification defines the property.

5.4 New Security Analysis Program

1) To prevent tampering with information

   The program information can effectively prevent tampering. Server A server key to return the device information, the device comprising two parts A wants to acquire, i.e. the request message and session key Ks before. Thus, the device A can know whether it's the original information is changed before the key server Server receives.

2) To prevent replay attacks

   The program can effectively prevent replay attacks. The random number Na key distribution stage may know whether the device A previous request information is reproduced. Random number Nb equipment can ensure the authentication stage device B session information has not been received replay attacks.

5.5 New Program Performance Analysis and Program Comparison

This section analyzes the performance of the new BACnet protocol device authentication. In the information interaction stage of the device, key distribution and identity authentication must be initiated. The new scheme not only USES the method of adding random Numbers to encrypt and verify messages, but also changes the way of key distribution. Therefore, the performance consumption of this part of the protocol has a great impact on the time cost of the whole communication. Most of the encryption methods used in key distribution and authentication use the cryptographic primitives in the BACnet specification, which do not require major upgrades to existing platforms, but do add some communication, computing, and storage overhead.

The comparison between the proposed scheme and the related BACnet protocol scheme is shown in table 6. Literature [8] made a detailed study on the identification problem, denial of service, eavesdropping and buffer overflow in the core functions of the protocol, and proposed deterministic improvement for BAS networking problems that were not taken into account at the beginning, and added the remote management technology of enterprise internal network and Internet connection. Literature [10] mainly discusses the limitations of secure communication and the security of data exchange in BAS. Holmberg *et al.* proposed mitigation measures for some of the identified vulnerabilities in BACnet, such as the BACnet firewall, which is computatively complex and requires dedicated hardware. Literature [13] identified the ability of legitimate malicious commands running within BACnet works to prevent them from transmitting data traffic through boundary firewalls, and proposed a potential solution for BAS specific intrusion detection systems (IDS).

Table 6: comparison table of this scheme and other BACnet protocol schemes

Attributes	[8]	[10]	[13]	Our scheme
DOS	✓	✓	✓	✓
Hacking	✓	-	-	✓
IDS	✓	-	✓	x
Information to	×	✓	×	✓
tamper with				
Replay attack	×	-	×	✓

6 Conclusion

In this paper, BACnet protocol device certification process as an object, to colored Petri net theory and Delov-Yao attack as a guide, based on CPN Tools model checking tools, focusing on formal modeling and safety assessment of the agreement, the agreement loophole mining is proposed targeted safety improvement program, and proposed a new program model checker application CPN tools for safety verification. BACnet protocol BACnet device authentication is only an agreement in the security services, other security services are lack of formal modeling and safety assessment. The next step, consider other security services are also studied.

Acknowledgments

This research is supported by The National Natural Science Foundation of China (No.61462060,No. 61762060) and The Network and Information Security Innovation Team of GansuProvincial Department of Education Lanzhou University of Technology (No.2017C-05). Tao Feng is the corresponding author.

References

[1] M. Bayat and M. R. Aref, "An attribute based key agreement protocol resilient to kci attack," *International Journal of Electronics & Information Engineering*, vol. 2, 2015.

[2] R. Casado-Vara, F. D. L. Prieta, S. Rodriguez, J. L. Calvo-Rolle, and J. Prieto, "Adaptive fault-tolerant tracking control algorithm for iot systems: Smart building case study," in *14th International Conference on Soft Computing Models in Industrial and Environmental Applications*, 2019.

[3] D. Fauri, M. Kapsalakis, D. R. dos Santos, E. Costante, and S. Etalle, "Leveraging semantics for actionable intrusion detection in building automation systems," in *13th International Conference on Critical Information Infrastructures Security*, 2018.

[4] R. Gansner, H. Reppy, "The standard ml basis manual," in *AT Corporation and Lucent Technologies Inc*, 2004.

[5] O. Gasser, Q. Scheitle, C. Denis, N. Schricker, and G. Carle, "Security implications of publicly reachable building automation systems," in *IEEE Security & Privacy Workshops*, 2017.

[6] W. Granzer and W. Kastner, "Communication services for secure building automation networks," in *IEEE International Symposium on Industrial Electronics*, 2010.

[7] D. G. Holmberg, "Bacnet wide area network security threat assessment," 2011.

[8] D. G. Holmberg, J. Bender, and M. Galler, "Using the bacnet; firewall router," *Ashrae Journal*, vol. 48, no. 11, pp. B10–B14, 2006.

[9] S. Hyun, J. Kim, H. Kim, J. Jeong, S. Hares, L. Dunbar, A. Farrel, "Interface to network security functions for cloud-based security services," *IEEE Communications Magazine*, vol. 56, no. 1, pp. 171–178, 2018.

[10] W. Kastner, G. Neugschwandtner, S. Soucek, and H. M. Newman, "Communication systems for building automation and control," pp. 1178–1203, 2005.

[11] J. Kaur, J. Tonejc, S. Wendzel, and M. Meier, *Securing BACnet's pitfalls*, Springer, 2016.

[12] S. E. Kim, J. P. Jeong, H. Ko, and H. Kim, "A flexible architecture for orchestrating network security functions to support high-level security policies," in *Proceedings of the 11th International Conference on Ubiquitous Information Management and Communication*, p. 5, 2017.

[13] N. Li, L. Chen, M.Chen, and Z. Lu, "A method of building safety rule base for power mobile terminals," pp. 72–75, 2018.

[14] L. Liu, L. Wang, and Z. Cao, "A note on one protocol for subset sum problem," *International Journal of Electronics & Information Engineering*, vol. 2, pp. 103–119, 2019.

[15] W. Mao, "A structured operational modelling of the dolev-yao threat model," in *Security Protocols*, pp. 34–46, 2004.

[16] H. M. Newman, "Bacnet; the global standard for building automation and control networks," *Business Expert Press*, 2013.

[17] M. Peacock, M. N. Johnstone, and J. I. D. Hartog, "Timing attack detection on bacnet via a machine learning approach," in *Australian Information Security Management Conference*, 2015.

[18] A. V. Ratzer, L. Wells, H. M. Lassen, M. Laursen, J. F. Qvortrup, M. S. Stissing, M. Westergaard, S. Christensen, and K. Jensen, "CPN tools for editing, simulating, and analysing coloured petri nets," in *Applications and Theory of Petri Nets*, pp. 450–462, 2003.

[19] O. N. Samijayani, L. Addien, I. Fauzi, and M. Zasyi, "Multi-sensing wireless sensor network for smart building system," *Journal of Computational & Theoretical Nanoscience*, vol. 23, no. 4, pp. 3660–3664, 2017.

[20] M. Weber, E. Kindler, "The petri net markup language," *Acta Simulata Systematica Sinica*, 2003.

[21] S. Wendzel, B.Kahler, and T.Rist, "Covert channels and their prevention in building automation protocols: A prototype exemplified using bacnet," in *IEEE International Conference on Green Computing and Communications*, pp. 731–736, 2012.

[22] H. Xinya, L. Jianhua, and L. Hao, "Construction and analysis of network security situation awareness model based on colored petri nets," *Computer and Digital Engineering*, vol. 47, no. 2, 2019.

[23] P. Čeleda, R. Krejčí, V. Krmíček, "Flow-based security issue detection in building automation and control networks," in *Information and Communication Technologies*, pp. 64–75, 2012.

Biography

FENG Tao, was born in 1970, researcher/PhD supervisor, CCF senior member, IEEE and ACM member. He graduated from Xidian University,and then worked at school of computer and communication in Lanzhou University of Technology. His research interests include Cryptography and information security. words.

Jiang Xiao-yan, was born in 1993, CCF member.She is a master's student at lanzhou university of technology. Her research interests include technical information security and industrial control systems.

Fang Jun-li, was born in 1985, CCF member. She is a doctor's student at lanzhou university of technology. Her research interests include technical information security and industrial control systems.

Gong Xiang, was born in 1986, CCF member. He is a doctor's student at lanzhou university of technology. His research interests include technical information security and industrial control systems.

A Note on One Outsourcing Algorithm for Modular Exponentiations

Lihua Liu and Bin Cheng
(Corresponding author: Lihua Liu)

Department of Mathematics, Shanghai Maritime University
Haigang Ave 1550, Shanghai, 201306, China
Email: liulh@shmtu.edu.cn

(Received Dec. 14, 2021; Revised and Accepted Oct. 11, 2022; First Online Oct. 15, 2022)

Abstract

Fu *et al.* have presented two outsourcing algorithms for modular exponentiations [Cluster Comp., 21(4), 2018, 1933–1947] by using a single untrusted cloud server. We find the schemes have some flaws: (1) the outsourcing mechanism is infeasible because it cannot bring the outsourcer significant cost savings; (2) the outsourcing of Schnorr signature fails to keep its consistency; (3) it confused the general signature model with the blind signature model. We hope this note could clarify some misunderstandings about outsourcing computation and blind signature.

Keywords: Blind Signature; Modular Exponentiation; Outsourcing Computation; Untrusted Server

1 Introduction

Outsourcing computation can shift operations, jobs, or processes to a third party for reducing costs or improving efficiency. While it has many advantages, it also presents some challenges. The relationship with the third party must be properly managed, which includes incurred communication costs and security issues. An outsourcing scheme must bring the outsourcer plenty of cost savings, and cannot leak any outsourcer's privacy.

In 2011, Dreier and Kerschbaum [10] put forth a method for secure outsourcing of linear programming. Wang *et al.* [22] proposed an outsourcing scheme for large-scale systems of linear equations, but its homomorphic encryption system [15] was not compatible with Jacobi iteration [2]. In 2013, Chen, Yang and Hwang [6] also discussed the problem of privacy protection data access control. Chen *et al.* [5] presented one algorithm for outsourcing linear regression problem, but neglected to check whether the client can solve the original problem solely [3]. In 2018, Salinas *et al.* [18] presented an outsourcing scheme for large-scale sparse linear systems of equations. Ding *et al.* [9] pointed out that in the Salinas *et al.*'s scheme the cloud server can recover a client's input. Wang *et al.* [12,23] have presented a survey for reversible data hiding for VQ-compressed images. Pan *et al.* [7, 14, 16, 17, 21] put forth some batch verification schemes for identifying illegal signatures, smart card-based password authentication schemes, and data collaboration scheme with hierarchical attribute-based encryption in cloud computing. Hwang, Lee and Tzeng [13] presented a new proxy signature scheme for a specified group of verifiers. Yang, Chang and Hwang [24] proposed a new group signature scheme based on RSA assumption.

Given the finite field $\mathbb{F}_p$ where p is a large prime, $g \in \mathbb{F}_p$, ℓ is a positive integer, the cost for computing $g^\ell \bmod p$ is $O(\log \ell \log^2 p)$. In practice, there are some elegant algorithms for the common computation [1,4]. If ℓ is very big, such a modular exponentiation could be expensive for some resource-constrained devices. Based on this observation, Fu *et al.* [11] have presented two outsourcing schemes for modular exponentiations.

In this note, we show that the schemes have flaws. In the proposed mechanism, we find, the outsourcer is actually capable of doing modular exponentiation solely. It is unnecessary for the user to outsource such computations in the discussed scenario. We also point out the loss of consistency in the outsourcing of Schnorr signature, and correct a misunderstanding about the primitive of blind signature.

2 Review of the Schemes

In the model, there are two entities: the user T and the untrusted cloud server U. The outsourcer T is the one with limited resources. The server U may be curious about the user's data and even cheat him.

Let p and q be two big primes, $q \mid p-1$, $a \in \mathbb{Z}_q^*, u \in \mathbb{Z}_p^*$. The user wants to securely outsource the computation of $u^a \bmod p$ to the server. The scheme can be described as follows (see Table 1), where *Rand* is a random pairs generating tool (a table stores some revelent pairs).

Denote the outsourcing algorithm for modular exponentiation by MExp, and its variation for multiple mod-

ular exponentiations by M2Exp. Based on these basic outsourcing algorithms, Fu *et al.* [11] have presented two applications for outsourcing Cramer-Shoup encryption [8] and Schnorr signature [19]. We now only focus on the outsourcing of Schnorr signature (see Table 2).

3 Security Analysis

3.1 The Artificial Outsourcing Mechanism

The scheme's correctness is due that

$$\begin{aligned} u^a \bmod p &= (v_1 w_1)^a = (g^{x_1})^a w_1^a = g^{x_2+r} w_1^{l_1+k_1 t_1} \\ &\xlongequal{\text{if } g^q \bmod p=1} g^{x_2} g^r w_1^{l_1} (w_1^{k_1})^{t_1}, \\ u^a \bmod p &= (v_2 w_2)^a = g^{x_3 a} w_2^a = g^{x_4+r'} w_2^{l_2+k_2 t_2} \\ &\xlongequal{\text{if } g^q \bmod p=1} g^{x_4} g^{r'} w_2^{l_2} (w_2^{k_2})^{t_2}, \end{aligned}$$

which means that $g \in \mathbb{Z}_p$ is of the order q. But the necessary condition is not specified. By the way, the checking equation (see Equation (5), §4.2, [11]),

$$g^{x_2} g^r w_1^{l_1} (w_1^{k_1})^{t_1} = g^{x_4} g^{r'} w_2^{l_2} (w_2^{k_2})^{t_2},$$

should be corrected as the congruence equation

$$g^{x_2} g^r w_1^{l_1} (w_1^{k_1})^{t_1} = g^{x_4} g^{r'} w_2^{l_2} (w_2^{k_2})^{t_2} \bmod p.$$

Notice that the user needs to send $\{(r'/x_5, g^{x_5}), (k_1, w_1), (k_2, w_2), (l_1, w_1), (l_2, w_2), (r/x_5, g^{x_5})\}$ to the server, while the server needs to return

$$\{g^r, g^{r'}, w_1^{l_1}, w_2^{l_2}, w_1^{k_1}, w_2^{k_2}\}$$

to the user. In the equation $l_1 = a - k_1 t_1 \bmod q$, both l_1 and k_1 are known to the server. The exponent a should be kept secret from the server. Hence, t_1 is inaccessible to the server. Likewise, in the equation $l_2 = a - k_2 t_2 \bmod q$, t_2 is also inaccessible to the server. Taking into account this fact, we conclude that the user has to solely compute $(w_1^{k_1})^{t_1} \bmod p$ and $(w_2^{k_2})^{t_2} \bmod p$ in Eq.(1). That is to say, the user wants to securely outsource $u^a \bmod p$ at the expense of solely computing other two modular exponentiations with the same modulus p. We refer to the comparisons in two different scenarios (Table 3).

Except the incurred communication cost (depending on the practical environment), the theoretical cost saving depends on the size of t_1 and t_2. Since t_1 and t_2 are randomly picked in order to protect the secret exponent a, it is hard to assert that $\log a \gg \log t_1 + \log t_2$. Even if both t_1 and t_2 are restricted to 64-bit numbers, and a is assumed to be of 1000 bits (see §5.3, [11]), the computational cost saving is about $O(870 \log^2 p)$. Practically, the modulus p is of 1024 bits. The arithmetical cost saving is nearly 0.005 seconds (on PC with Intel(R) Core(TM) i7-4790, 3.60GHz, 4GB RAM). Frankly, the saving is quite negligible.

3.2 Inconsistencies

The outsourcing scheme fails to keep its consistency.

◇ $s = k + xe \bmod q$, $g^s g^{-e} = g^{k+xe-e} \neq g^k \bmod p$. Hence, Alice should invoke M2Exp$(s, -e; g, y)$ and check that $r = g^s y^{-e} \bmod p$, instead of $r = g^s g^{-e} \bmod p$.

◇ The checking equation for the verifier is incorrect.

$$\begin{aligned} R &= r g^\alpha g^\beta = g^{k+\alpha+\beta} \bmod p, \\ R' &= g^S g^{-E} = g^{S-E} = g^{k+xe+\beta-(e-\gamma)} \\ &= g^{k+\gamma+\beta+xe-e} \neq R \bmod p. \end{aligned}$$

Even if γ is set as α, it is still incorrect. To revise, we could specify that $R = r y^\alpha g^\beta = g^{k+x\alpha+\beta} \bmod p$, and

$$R' = g^S y^{-E} = g^{k+xe+\beta-x(e-\alpha)} = g^{k+x\alpha+\beta} \bmod p.$$

◇ The necessity of outsourcing has been casually overstated [3]. So did the scheme. The order of g modulo p is q, i.e., $g^q = 1 \bmod p$. Hence, the modular exponentiation $g^k \bmod p$ where $k \in \mathbb{Z}_p^*$ is picked by Bob, is actually equivalent to $g^{k'} \bmod p$, where $k' = k \bmod q$. In view of that $q \leq 2^{160}$, the computational cost for $g^{k'} \bmod p$ is very negligible. In this case, the outsourcing scheme becomes insignificant.

3.3 A Misunderstanding

Note that the verifier eventually checks

$$E = E' \Longleftrightarrow E = h(R' \| M) \Longleftrightarrow E = h(g^S y^{-E} \| M),$$

where g, y are two public parameters, and S, E are generated by Alice with the help of Bob. The scheme is not a general signature, instead a blind signature [20], in which the true signer (Bob) cannot link the final signature (S, E) to the real requestor (Alice), because $E = e - \alpha \bmod q, S = s + \beta \bmod q$. Both α and β are randomly chosen by Alice, and never disclosed. The work has confused the general signature model with blind signature model.

4 Conclusion

We show that the Fu *et al.*'s outsourcing scheme cannot bring the outsourcer plenty of cost savings, because the outsourcer himself can do the general modular exponentiations. We also show that there are some inconsistencies. We would like to stress that the necessity of outsourcing such common computations should be carefully investigated and balanced.

Acknowledgements

We thank the National Natural Science Foundation of China (Project 61411146001). We are grateful to the reviewers for their valuable suggestions.

Table 1: Fu *et al.*'s outsourcing scheme for modular exponentiation

User: $\{u, a, q, p\}$	Server
Invoke *Rand* to generate the blinding pairs $\{(x_i, g^{x_i} \bmod p)\}_{1\le i\le 5}$, where $g \in \mathbb{Z}_p^*$. Set $v_1 = g^{x_1} \bmod p, v_2 = g^{x_3} \bmod p$. Compute $w_1 = u/v_1 \bmod p, w_2 = u/v_2 \bmod p$, $r = x_1 a - x_2 \bmod q, r' = x_3 a - x_4 \bmod q$. Pick k_1, t_1, k_2, t_2, and compute $l_1 = a - k_1 t_1 \bmod q, l_2 = a - k_2 t_2 \bmod q$. $\xrightarrow[(r'/x_5, g^{x_5}),(k_1,w_1),(k_2,w_2)]{(l_1,w_1),(l_2,w_2),(r/x_5,g^{x_5})}$	Compute $g^r = (g^{x_5})^{r/x_5} \bmod p$, $g^{r'} = (g^{x_5})^{r'/x_5} \bmod p$, $w_1^{l_1} \bmod p$, $w_2^{l_2} \bmod p, w_1^{k_1} \bmod p, w_2^{k_2} \bmod p$. $\xleftarrow[w_1^{l_1}, w_2^{l_2}, w_1^{k_1}, w_2^{k_2}]{g^r,\ g^{r'}}$
Check that $g^{x_2} g^r w_1^{l_1} (w_1^{k_1})^{t_1} = g^{x_4} g^{r'} w_2^{l_2} (w_2^{k_2})^{t_2} \bmod p$. If true, output $g^{x_2} g^r w_1^{l_1} (w_1^{k_1})^{t_1}$.	

Table 2: The outsourcing of Schnorr signature

Alice (message M)	Cloud server	Bob (signer)	Carly (verifier)
		Pick primes p, q, $q \mid q-1$, $p \ge 2^{512}, q \le 2^{160}$. Select $g \in \mathbb{Z}_p^*$, $g^q = 1 \bmod p$. Pick $x \in \mathbb{Z}_q$, and set $y = g^x \bmod p$. Publish $\{p, q, g, y\}$ and the hash function h.	
	$\xrightarrow[\text{[to Bob]}]{\text{MExp}(k,g)}$	Pick $k \in \mathbb{Z}_p^*$, invoke $\text{MExp}(k, p)$. Set $r = g^k \bmod p$.	
Pick $\alpha, \beta \in \mathbb{Z}_p^*$, and invoke $\text{M2Exp}(\alpha, \beta; g, g)$.	$\xleftarrow[\text{[to Alice]}]{\text{M2Exp}(\alpha,\beta;g,g)}$	$\xleftarrow[\text{[to Alice]}]{r}$	
Set $R = rg^\alpha g^\beta \bmod p$, $E = h(R\|M), e = E + \gamma \bmod q$. $\xrightarrow[\text{[to Bob]}]{e}$		Set $s = k + xe \bmod q$. $\xleftarrow[\text{[to Alice]}]{s}$	
Invoke $\text{M2Exp}(s, -e; g, g)$.	$\xleftarrow[\text{[to Alice]}]{\text{M2Exp}(s,-e;g,g)}$		
Check $r = g^s g^{-e} \bmod p$. If true, set $S = s + \beta \bmod q$. $\xrightarrow[\text{[to Carly]}]{(E,S),M}$	$\xrightarrow[\text{[to Carly]}]{\text{M2Exp}(S,-E;g,g)}$		Invoke $\text{M2Exp}(S, -E; g, g)$. Set $R' = g^S g^{-E} \bmod p$, $E' = h(R'\|M)$. Check that $E' = E$.

Table 3: The dominated computations for the user in two different scenarios

The general scenario	The outsourcing scenario
	Lookup the table *Rand* for pairs $\{(x_i, g^{x_i} \bmod p)\}_{1\le i\le 5}$. —— $O(1)$ Inverse computations for $w_1 = u/v_1 \bmod p, w_2 = u/v_2 \bmod p$. —— $O(\log^2 p)$ $r = x_1 a - x_2 \bmod q, r' = x_3 a - x_4 \bmod q$. $l_1 = a - k_1 t_1 \bmod q, l_2 = a - k_2 t_2 \bmod q$. —— $O(\log^2 q)$ Compute $g^{x_2} g^r w_1^{l_1} (w_1^{k_1})^{t_1} \bmod p$. —— $O(\log t_1 \log^2 p)$
$u^a \bmod p$. ——$O(\log a \log^2 p)$	Compute $g^{x_4} g^{r'} w_2^{l_2} (w_2^{k_2})^{t_2} \bmod p$. ——$O(\log t_2 \log^2 p)$

References

[1] P. Barrett, "Implementing the Rivest Shamir and Adleman public key encryption algorithm on a standard digital signal processor," in *Proceedings of 6th Annual Cryptology Conference, Advances in Cryptology - CRYPTO 1986*, pp. 311–323, Santa Barbara, USA, Aug. 1987.

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

[3] Z. Cao, L. Liu, and O. Markowitch, "Comment on 'highly efficient linear regression outsourcing to a cloud'," *IEEE Transactions on Cloud Computing*, vol. 7, no. 3, p. 893, 2019.

[4] Z. Cao and X. Wu, "An improvement of the Barrett modular reduction algorithm," *International Journal of Computer Mathematics*, vol. 91, no. 9, pp. 1874–1879, 2014.

[5] F. Chen and *et al.*, "Highly efficient linear regression outsourcing to a cloud," *IEEE Transactions on Cloud Computing*, vol. 2, no. 4, pp. 499–508, 2014.

[6] M. Y. Chen, C. C. Yang, and M. S. Hwang, "Privacy protection data access control," *International Journal of Network Security*, vol. 15, no. 6, pp. 411–419, 2013.

[7] Y.H. Chen and *et al.*, "Research on the secure financial surveillance blockchain systems," *International Journal of Network Security*, vol. 22, no. 4, pp. 708–716, 2020.

[8] R. Cramer and V. Shoup, "A practical public key cryptosystem provably secure against adaptive chosen ciphertext attack," in *Proceedings of 18th Annual International Cryptology Conference, Advances in Cryptology - CRYPTO 1998*, pp. 13–25, Santa Barbara, USA, August 1998.

[9] Q. Ding, G. Weng, G. Zhao, and C. Hu, "Efficient and secure outsourcing of large-scale linear system of equations," *IEEE Transactions on Cloud Computing*, vol. 9, no. 2, pp. 587–597, 2021.

[10] J. Dreier and F. Kerschbaum, "Practical privacy-preserving multiparty linear programming based on problem transformation," in *Proceedings of IEEE International Conference on Privacy, Security, Risk, and Trust, and IEEE International Conference on Social Computing*, pp. 916–924, Boston, USA, Oct. 2011.

[11] A. Fu and *et al.*, "Secure outsourcing algorithms of modular exponentiations with optimal checkability based on a single untrusted cloud server," *Cluster Computing*, vol. 21, no. 4, pp. 1933–1947, 2018.

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

[13] M. S. Hwang, C. C. Lee, and S. F. Tzeng, "A new proxy signature scheme for a specified group of verifiers," *Information Sciences*, vol. 227, pp. 102–115, 2013.

[14] L. H. Liu and Y. Liu, "On the anonymity of one multiserver authenticated key agreement with offline registration centre," *International Journal of Electronics and Information Engineering*, vol. 13, no. 3, pp. 105–110, 2021.

[15] P. Paillier, "Public-key cryptosystems based on composite degree residuosity classes," in *Proceeding of International Conference on the Theory and Application of Cryptographic Techniques, Advances in Cryptology - EUROCRYPT 1999*, pp. 223–238, Prague, Czech Republic, May 1999.

[16] H. T. Pan, E. F. Cahyadi, S. F. Chiou, and M. S. Hwang, "Research on batch verification schemes for identifying illegal signatures," *International Journal of Network Security*, vol. 21, no. 6, pp. 1062–1070, 2019.

[17] H. T. Pan, H. W. Yang, and M. S. Hwang, "An enhanced secure smart card-based password authentication scheme," *International Journal of Network Security*, vol. 22, no. 2, pp. 358–363, 2020.

[18] S. Salinas, C. Luo, X. Chen, W. Liao, and P. Li, "Efficient secure outsourcing of large-scale sparse linear systems of equations," *IEEE Transactions on Big Data*, vol. 4, no. 1, pp. 26–39, 2018.

[19] C. Schnorr, "Efficient signature generation for smart cards," in *Proceedings of 9th Annual International Cryptology Conference, Advances in Cryptology - CRYPTO 1989*, pp. 239–252, Santa Barbara, USA, Aug. 1989.

[20] C. Schnorr, "Enhancing the security of perfect blind DL-signatures," *Information Sciences*, vol. 176, no. 10, pp. 1305–1320, 2006.

[21] W. L. Tai, Y. F. Chang, and W. H. Huang, "Security analyses of a data collaboration scheme with hierarchical attribute-based encryption in cloud computing," *International Journal of Network Security*, vol. 22, no. 2, pp. 212–217, 2020.

[22] C. Wang, K. Ren, J. Wang, and Q. Wang, "Harnessing the cloud for securely outsourcing large-scale systems of linear equations," *IEEE Transactions on Parallel and Distributed Systems*, vol. 24, no. 6, pp. 1172–1181, 2013.

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

[24] C. C. Yang, T. Y. Chang, and M. S. Hwang, "A new group signature scheme based on RSA assumption," *Information Technology and Control*, vol. 42, no. 1, pp. 61–66, 2013.

Biography

Lihua Liu, associate professor, with Department of Mathematics, Shanghai Maritime University, received her PhD degree in applied mathematics from Shanghai Jiao Tong University. Her research interests include combinatorics and cryptography.

Bin Cheng is currently pursuing his master degree from Department of Mathematics, Shanghai Maritime University. His research interests include combinatorics and cryptography.

Security Analysis of TBPKI-2 Protocol Based on Minimal Element Theory

Wen-Bei Zong[1] and Lei Yu[1,2,3]
(Corresponding author: Lei Yu)
School of Computer Science and Technology, Huaibei Normal Universtiy[1]
School of Information and Contral Engineering, China University of Mining and Technology[2]
Anhui Big-Data Research Center on University Management[3]
Huaibei, Anhui 235000, China
Email: yulei@chnu.edu.cn
(Received May 23, 2022; Revised and Accepted Oct. 11, 2022; First Online Oct. 15, 2022)

Abstract

The strand space model is a hybrid proof method combining theorem proof and protocol trace. It can not only analyze the correctness of security protocol but also be used to construct an attack model and reveal the internal defects of security protocol. Compared with other branches of the theory, the minimal element theory has more detailed and adequate advantages in the process of protocol analysis. For example, the TBPKI-2 protocol is a wireless network authentication protocol. It is optimized based on the PKI mechanism and has specific practical significance. Therefore, based on strand space theory, this paper analyzes the confidentiality and consistency of the protocol by using minimal element theory, accurately finds that the potential hidden danger in the protocol and its root cause is unable to block Man-in-the-Middle Attack, and proposes corresponding improvement suggestions according to the hidden danger and its root cause.

Keywords: Security Analysis; Security Protocol; Strand Space; TBPKI-2 Protocol

1 Introduction

Security protocol is a cryptography-based protocol that provides a variety of security services. With the rapid development of networking and information technology, some widely used protocols have gradually revealed their shortcomings. Therefore, it is necessary to analyze their security before improving or designing new security protocols. At present, there are mainly two analysis methods: non-formal and formal. Among the many formal analysis methods [8,11,17], in 1977, Fabrega, Herzog and Guttman established the strand space model theory [10], which is widely respected for its efficiency and rigor, simplicity and intuitiveness, and scalability, pushing the formal analysis technique of security protocol to a new level.

In recent years, it has been widely used in the analysis of security protocols [5, 7, 12, 16]. Strand space is a method combining theorem proof and protocol tracking. It can not only prove the correctness of security protocols, but also construct attacks and reveal the inherent defects of security protocols. With continuous research, strand space theory has been improving and expanding [9, 13]. Since the establishment of strand space model, there are three theoretical branches, namely, ideal and honesty, minimal element and authentication tests. Compared with other theoretical branches, the minimal element theory is more detailed and sufficient in the process of protocol analysis [15]. With the development of science and technology, the security of key agreement protocol in wireless communication [2, 4] has been attracting extensive attention. Therefore, in order to ensure the security of the protocol, we must analyze its security before using it. TBPKI-2 protocol [1] is a wireless network authentication protocol based on CVT. CVT is the validity credential of the entity's public key certificate. And the certificate ID of the entity, the validity term of the CVT and the public key of the entity can be decrypted from it. The content of the protocol is that A confirms its identity by showing CVT to B and completes the key negotiation between A and B.

Based on the minimal element theory in strand space, this paper will make a formal analysis of TBPKI-2 protocol from two aspects of confidentiality and consistency, point out the internal defects of the protocol and put forward some suggestions for improvement.

2 Strand Space Model Theory

2.1 Basic Concepts

Strand space is a two-tuple (Σ, tr), where Σ represents a set of strands. And strands among Σ can be used to

represent any sequence, tr represents a mapping of the sequence composed of elements from Σ to A. Some basic concepts in strand space are given below (the basic concepts and theorems of minimal element theory can be found in [3,10]):

1) Node n is a two-tuple $< s, i >$, where s is an element in Σ and i represents the sequence number of the node on this strand. Each node belongs to a unique strand. Node set is marked as N.

2) If $n =< s, i >$, the participant action represented by this node is represented as $(tr(s))_i = R_a$, where R_+ or $\_$, and a represents a message, then the node means that the participant sends or receives a.

3) If $n_1, n_2 \in N$, definition $n_1 \rightarrow n_2$ means $n_1 = +a, n_2 = -a$, which indicates that the message is sent from n_1 to n_2.

4) If $n_1, n_2 \in N$, definition $n_1 \Rightarrow n_2$ means that n_1 and n_2 are on the same strand and n_2 is the next node of n_1.

5) An unsigned term t appears in $n \in N$ if and only if $t \sqsubset term(n)$.

6) Let I as an unsigned term set. Node $n \in N$ is the entry point of I if and only if $term(n) = +t$, where $t \in I$, and for all nodes $n^{'} \Rightarrow^{+} n$, there is $term(n) \notin I$.

7) The unsigned term t originates from node $n \in N$ if and only if n is the entry point of the set $I = \{t^{'} : t \sqsubset t^{'}\}$.

8) The unsigned term t is uniquely originated if and only if t originates from the unique node $n \in N$.

Lemma 1. *Let C be a bundle, then $\preceq_c$ is a partial order relation with self-reflexivity, antisymmetry and transitivity. Any nonempty subset of bundle C has minimal elements under the partial order relation $\preceq_c$.*

Lemma 2. *Let C be a bundle and $S \subseteq C$ as a set of nodes satisfies the following property: $\forall m, m^{'}, unsterm(m) = unsterm(m^{'})$. Then $m \in S$ if and only if $m^{'} \in S$.*

If n is a $\preceq_{c-}$ minimal element of S, the sign of n is positive.

2.2 Penetrator Capability Description

In strand space theory, the penetrator's abilities are described by two parts: one is the key set initially mastered by the penetrator, and the other is the new information generated by the message that penetrator has intercepted. The atomic behavior of the penetrator is described by the penetrator trace, which is defined below:

1) M message: $< +t >$, where $t \in T$.

2) K key: $< +K >$, where $K \in K_p$.

3) C connect: $< -g, -h, +gh >$.

4) S separation: $< -gh, +g, +h >$.

5) E encryption: $< -K, -h, +\{h\}_K >$.

6) D decryption: $< -K^{-1}, -\{h\}_K, +h >$.

Definition 1. *Infiltrated strand space is a two-tuple (Σ, tr), where Σ is a strand space and $P \subseteq \Sigma$ satisfies the following condition: for all $p \subseteq P$, $tr(p)$ is a penetrator strand.*

Strands in P are called penetrator strands. Thus, if $s \in P$, strand $s \in \Sigma$ is a penetrator strand. And if strand is a penetrator strand, node n is called penetrator node. In addition, all strands and nodes are called regular strands and regular nodes.

Proposition 1. *Let C be a bundle and $K \in K \setminus K_p$.*

If K does not originate from a regular node, $K \not\sqsubset term(n)$ holds for any node $n \in C$. Specially, for any penetrator node $p \in C$, there is $K \not\sqsubset term(p)$.

3 Symbols and Assumptions

3.1 Symbols

The symbols used in this paper and their semantics are shown in Table 1.

3.2 Assumptions

The following assumptions are consistent with the actual situation.

1) Legitimate subjects in the network can also launch attacks;

2) The random number N_a is chosen irrelevantly to N_b. It can be proved that they are almost impossible to be equal in the probability model.

4 Strand Space Model and Analysis of TBPKI-2 Protocol

Further concretizing the term algebra:

1) Identifier set: $T_{name} \subseteq T$.Generally, A, B ... is used to represent identifier of the subject;

2) Mapping: $T_{name} \rightarrow K$. This mapping binds the subject to its public key.

The protocol is as follows:

1) $A \rightarrow B : CVT_A, N_a, K_i, Sign_a$.
$Sign_a = \{CVT_A, N_a, K_i\}_{K_a^{-1}}$, indicates the signature of subject A for this message;

2) $B \rightarrow A : CVT_B, N_a, \{N_b\}_{K_{ab}}, K_r, Sign_b$.
$Sign_b = \{CVT_B, N_a, \{N_b\}_{K_{ab}}, K_r\}_{K_b^{-1}}$, indicates the signature of subject B for this message;

3) $A \rightarrow B : \{N_b - 1\}_{K_{ab}}$.

Table 1: The semantics of symbols in the paper

Symbols	Semantics of symbols
A	Initiator of the protocol.
B	Responder of the protocol.
P	Penetrator of the protocol.
K_a, K_b, K_p	Public key of subject A, subject B and subject P.
K_a^{-1}, K_b^{-1}	Private key of subject A, subject B and subject P.
N_a, N_b	Random number generated by subject A and subject B.
CVT_a, CVT_b	Validity certificate of public key certificates of subject A and subject B.
$TCVP$	Trusted and valid third party.
(g, n)	The public number of D-H algorithm [6], and g is the primitive element of module n.
K_i, K_r	The partial key generated by subject A and subject B (g^x, g^y).
K_{ab}	Session keys for subjects A and B (g^{xy}).
C	Bundle.
Σ	Strand space.
$a \sqsubset b$	Term a is a subterm of term b.
s	Strand.
$Sign_a, Sign_b$	Signatures made with private keys K_a^{-1} and K_b^{-1} of subject A and subject B.

4.1 Strand Space of TBPKI-2

Definition 2. *Let (Σ, P) be an infiltrated strand space. If Σ is composed of the following three strands, it is called a TBPKI-2 strand space.*

1) Penetrator strand $s \in P$;

2) Initiator strand $s \in Init[A, B, N_a, N_b, CVT_A, CVT_B, K_i, K_r]$. Its trace is $< +CVT_A N_a K_i Sign_a, -CVT_B N_a \{N_b\}_{K_{ab}} K_r Sign_b, +\{N_b - 1\}_{K_{ab}} >$. Here $A, B \in T_{name}$, $N_a, N_b \in T$ and $N_a \notin T_{name}$. $Init[A, B, N_a, N_b, CVT_A, CVT_B, K_i, K_r]$ represents the set of all strands having above trace, and the subject corresponding to this strand is A;

3) Responder strand $s \in Resp[A, B, N_a, N_b, CVT_A, CVT_B, K_i, K_r]$ is corresponding to the initiator strand. Its trace is $< -CVT_A N_a K_i Sign_a, +CVT_B N_a \{N_b\}_{K_{ab}} K_r Sign_b, -\{N_b - 1\}_{K_{ab}} >$. Here $A, B \in T_{name}$, $N_a, N_b \in T$ and $N_b \notin T_{name}$. $Resp[A, B, N_a, N_b, CVT_A, CVT_B, K_i, K_r]$ represents the set of all strands having above trace, and the subject corresponding to this strand is B.

If $s \in Init[A, B, N_a, N_b, CVT_A, CVT_B, K_i, K_r]$ is a regular strand, A is called the initiator of s. And if $s \in Resp[A, B, N_a, N_b, CVT_A, CVT_B, K_i, K_r]$ is a regular strand, B is called the responder of s. N_a, N_b are called the corresponding initiator and responder values.

4.2 Responder Analysis for TBPKI-2 Protocol

4.2.1 Consistency Analysis of Responder

Proposition 2. *Assuming the following conditions are valid:*

1) Σ is a TBPKI-2 space, C is a bundle of Σ, s is a responder strand. And $s \in Resp[A, B, N_a, N_b, CVT_A, CVT_B, K_i, K_r]$, with $C - hight(s) = 3$;

2) $K_b \notin K_p$;

3) $N_a \neq N_b$, and N_b is the only origin in Σ.

Therefore, C contains an initiator strand $t \in Init[A, B, N_a, N_b, CVT_A, CVT_B, K_i, K_r]$, with $C - hight(t) = 3$.

Arbitrarily Select $\Sigma, C, s, A, B, N_a, N_b, CVT_A, CVT_B, K_i, K_r$ that satisfies the assumptions in Proposition 2. Node $< s, 2 >$ outputs the value $CVT_B N_a \{N_b\}_{K_{ab}} K_r Sign_b$. It is marked as n_0, and its term is marked as v_0. Node $< s, 3 >$ receives the value $\{N_{b-1}\}_{K_{ab}}$. It is marked as n_3 and its term is marked as v_3. In the proof process, other two nodes n_1 and n_2 are used, which satisfy $n_0 \prec n_1 \prec n_2 \prec n_3$.

Lemma 3. *N_b originates from n_0.*

Proof. By Proposition 1, $N_b \sqsubset v_0$, and the sign of n_0 is positive. Therefore, it only need to prove $N_b \not\sqsubset n^{'}$, where $n^{'}$ is the precursor node $< s, 1 >$ on the same strand as n_0. By Proposition 2, $N_a \neq N_b$ can be proved, so $term(n^{'}) = \{CVT_A, N_a, K_i, Sign_a\}$. Finally, it need to verify $N_b \neq A$. By Definition 1, $N_b \notin T_{name}$, so $N_b \neq A$. Thus, $N_b \not\sqsubset n^{'}$. □

Lemma 4. *Set $S = \{n \in C : N_b \sqsubset term(n) \bigwedge v_0 \not\sqsubset term(n)\}$ has a minimal element $\preceq_{n2}$, n_2 is a regular node and its sign is positive.The initiator strand contains nodes n_1 and n_2, and the responder strand contains nodes n_0 and n_3. Node n_2 contains N_b.*

Proof. Because $n_3 \in C$ and n_3 contains N_b but not v_0, $n_3 \in S$. Therefore, it is a nonempty set. By Lemma 1, there is at least one minimal element $\preceq_{-n2}$. By Lemma 2, the sign of n_2 is positive.

According to the trace of penetrator strand P, it is proved that n_2 cannot be on penetrator strand P.

M: Trace $tr(p)$ has form $< +t >$, where $t \in T$. Thus, $t = N_b$. At this time, N_b originates from this strand, but this is obviously impossible. By lemma 3, N_b originates from a regular node n_0, and according to Assumption 3 of Proposition 2, N_b is the only origin in Σ. Therefore, N_b is not generated on the strand M;

C: Trace $tr(p)$ has form $< -g, -h, +gh >$. It is obvious that the regular node is not the minimal element of set S. Therefore, N_b is not generated on strand C;

K: Trace $tr(p)$ has form $< +K_0 >$, where $K_0 \in K_p$. But $N_b \not\sqsubset K_0$. Therefore, N_b is not generated on strand K;

E: Trace $tr(p)$ has form $< -K_0, -h, +\{h\}_{K_0} >$, assuming $N_b \sqsubset \{h\}_{K_0} \bigwedge v_0 \not\sqsubset \{h\}_{K_0}$. Because $N_b \neq \{h\}_{K_0}$, there is $N_b \sqsubset h$. However, $v_0 \not\sqsubset h$, so this positive node cannot be the minimal element of set S. Therefore, N_b is not generated on strand E;

D: Trace $tr(p)$ has form $< -K_0^{-1}, -\{h\}_{K_0}, +h >$. If this positive node is the minimal element of set S, then there must exist $v_0 \not\sqsubset h$ and $v_0 \sqsubset \{h\}_{K_0}$. Therefore, according to the free encryption assumption, there must be $h = \{CVT_B N_a \{N_b\}_{K_{ab}} K_r Sign_b\}$ and $K_0 = K_b^{-1}$. So there exists a node m (the first node on this strand) with $term(m) = K_b$. Because Proposition 1 assumes $K_b \notin K_p$, it is deduced that K_b originates from a regular node. But there is no initiator strand or responder strand originated from K_b. Therefore, N_b is not generated on strand D;

S: Trace $tr(p)$ has form $< -gh, +g, +h >$, assuming $term(n_2) = g$, which can be proved similarly when $term(n_2) = h$. $N_b \sqsubset g$ and $v_0 \not\sqsubset g$ due to $n_2 \in S$. From the minimality of n_2, there is $v_0 \sqsubset gh$. But $v_0 \neq gh$, so $v_0 \sqsubset h$.

Let $T = \{m \in C : m \prec n_2 \bigwedge gh \sqsubset term(m)\}$, each element in T is a penetrator node. Because regular node does not contain the subterm gh, and $< p, 1 > \in T$, T is a nonempty set. By Lemma 1,2, T contains a minimal element m, and its sign is positive. The following proof that m is impossible on penetrator strand S.

Firstly, the minimal element in T cannot appear on the strand of type M and K.

S: If $gh \sqsubset term(m)$, m is a regular node that lies on a S-type penetrator strand $p^{'}$. There is $gh \sqsubset term(< p^{'}, 1 >)$. And $< p^{'}, 1 > \prec m$ contradicts the minimality of m in T.

E: If $gh \sqsubset term(m)$, m is a regular node that lies on a E-type penetrator strand $p^{'}$. There is $gh \sqsubset term(< p^{'}, 2 >)$. And $< p^{'}, 2 > \prec m$ contradicts the minimality of m in T.

D: If $gh \sqsubset term(m)$, m is a regular node that lies on a D-type penetrator strand $p^{'}$. There is $gh \sqsubset term(< p^{'}, 2 >)$. And $< p^{'}, 1 > \prec m$ contradicts the minimality of m in T.

C: If $gh \sqsubset term(m)$, m is a regular node that lies on a C-type penetrator strand $p^{'}$, and m is the minimal element of T. Therefore, $gh = term(m)$, and the trace of $p^{'}$ has form $< -g, -h, +gh >$. So $term(< p^{'}, 1 >) = term(n_2)$. And $< p^{'}, 1 > \prec n_2$ contradicts the minimality of n_2 in S.

As mentioned above, n_2 cannot be on a penetrator strand, it must be on a regular strand. □

Definition 3. *Minimal element $\preceq n_2$ in the fixed set $S = \{n \in C : N_b \sqsubset term(n) \bigwedge v_o \not\sqsubset term(n)\}$. At this time, node n_2 is a regular node and its sign is positive.*

Lemma 5. *There exist a precursor node n_1 of node n_2 on strand t, and $term(n_1) = \{CVT_B, N_a, \{N_b\}_{K_{ab}}, K_r, Sign_b\}$. The lemma content is shown in Figure 1.*

Proof. By Lemma 3, N_b originates from n_0. According to Condition 3 of Proposition 2, N_b is the only origin in Σ. Because $v_0 \sqsubset term(n_0) \bigwedge v_0 \not\sqsubset term(n_2)$, $n_2 \neq n_0$. Thus, N_b does not originate from n_2. Because there is a precursor node n_1 of n_2 on strand t, $N_b \sqsubset term(n_1)$. From the minimality of n_2, it follows that $v_0 = \{CVT_B, N_a, \{N_b\}_{K_{ab}}, K_r, Sign_b\} \sqsubset term(n_1)$. From Assumptions 2 of Proposition 2, $K_b \notin K_p$, so $term(n_1) = \{CVT_B, N_a, \{N_b\}_{K_{ab}}, K_r, Sign_b\}$. □

Lemma 6. *The regular strand t containing n_1 and n_2 is an initiator strand of bundle C.*

Proof. Node n_2 is a regular node with positive sign and its precursor node n_1 has form $\{CVT_B N_a \{N_b\}_{K_{ab}} K_r Sign_b\}$. If t is a responder strand, it only be a node with negative symbol after n_1, so t is an initiator strand. Therefore, n_1

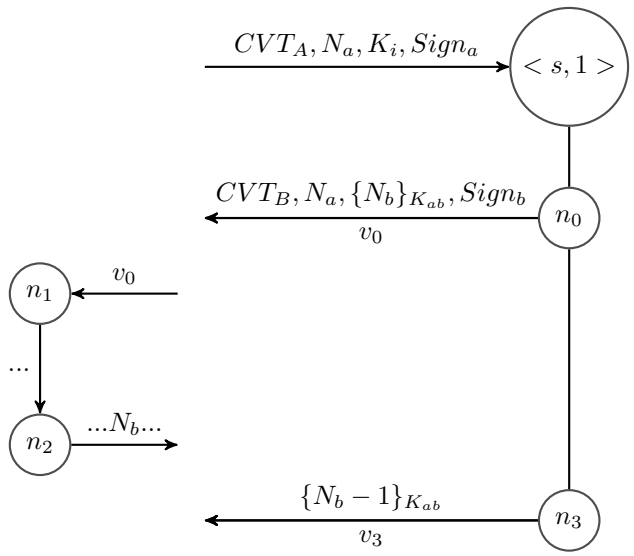


Figure 1: Node n_1 contains v_0

and n_2 are the 2nd and 3rd nodes on the strand respectively. The last node in t is contained in the bundle, so $C - hight(t) = 3$. □

Proposition 3. *Set Σ is a TBPKI-2 space, and N_a is the only origin in Σ. Therefore, for any A, B and N_b, there exist one such strand $t \in Init[A, B, N_a, N_b, CVT_A, CVT_B, K_i, K_r]$ at most.*

Proof. For any A, B, N_a, if $t \in Init[A, B, N_a, N_b, CVT_A, CVT_B, K_i, K_r]$, the sign of $< t, 1 >$ is positive, $N_a \sqsubset term(< t, 1 >)$ and N_a cannot appear earlier on t. Therefore, N_a originates from node $< t, 1 >$. Thus, if N_a is the only origin in Σ, there exist one such t at most. □

4.2.2 Confidentiality Analysis of Responder

Proposition 4. *Assuming the following conditions are valid:*

1) Σ is a TBPKI-2 space, C is a bundle of Σ, s is a responder strand. And $s \in Resp[A, B, N_a, N_b, CVT_A, CVT_B, K_i, K_r]$, with $C - hight(s) = 3$;

2) $K_a \notin K_p$, and $K_b \notin K_p$;

3) $N_a \neq N_b$, and N_b is the only origin in Σ.

Therefore, for any node $m \in C$ satisfying $N_b \sqsubset term(n)$, $\{CVT_B N_a \{N_b\}_{K_{ab}} K_r Sign_b\} \sqsubset term(m)$ is established or $\{N_b - 1\}_{K_{ab}} \sqsubset term(m)$ is established. Specially, $N_b \neq term(m)$.

Arbitrarily select Σ, C, s, A, B, N_a, N_b, CVT_A, CVT_B, K_i, K_r that satisfies the assumptions in Proposition 2. Node $< s, 2 >$ outputs the value $CVT_B N_a \{N_b\}_{K_{ab}} K_r Sign_b$. It is marked as n_0, and its term is marked as v_0. Node $< s, 3 >$ receives the value $\{N_b - 1\}_{K_{ab}}$. It is marked as n_3, and its term is marked as v_3. Consider the following set: $S = \{n \in C : N_b \sqsubset term(n) \bigwedge v_0 \not\sqsubset term(n) \bigwedge v_3 \not\sqsubset term(n)$.

Lemma 7. *The minimal element of S is not a regular node.*

Proof. Inversely assumed that there exist a minimal element that is a regular node $m \in S$. According to Lemma 2, the sign of m is positive.

1) Only the sign of n_0 is positive and $v_0 \sqsubset term(n_0)$, so m cannot be on the strand s;

2) Assume that is located on the responder strand $s' \neq s$. Then, $m =< s', 2 >$, $term(n) = \{CVT, N, \{N'\}_{K_d}, K, Sign_e\}$. Because $N_b \sqsubset term(m)$, $N_b = N$ or $N_b = N'$.

   a. If $N_b = N$, because the term of $< s', 1 >$ is $\{CVT, N, K, Sign_c\} = \{CVT, N_b, K, Sign_c\}$, $N_b \sqsubset term(< s', 1 >)$. And $v_0 \not\sqsubset \{CVT, N_b, K, Sign_c\}$, $v_3 \not\sqsubset \{CVT, N_b, K, Sign_c\}$, so $< s', 1 > \in S$. However, $< s', 1 > \prec m$ contradicts the minimality of m;

   b. If $N_b \neq N$ and $N_b = N'$, so N_b originates from m. It contradicts that n_0 is the only origin of N_b.

So m cannot be on responder strand $s' \neq s$.

1) Assuming it is located on the initiator strand $s' \neq s$. Then m may be located at the 1st node or the 3rd node of s'.

   a. If $m =< s', 1 >$, because $N_b \sqsubset term(m)$, N_b originates from m. It contradicts that n_0 is the only origin of N_b;

   b. If $m =< s', 3 >$, $term(m) = \{N_b - 1\}_{K_{ab}}$, the second node $< s', 2 >$ has the form $\{CVT, N, \{N_b\}_{K_d}, K, Sign_e\}$. It contradicts the minimality of m.

So m cannot be on initiator strand $s' \neq s$. □

Lemma 8. *The minimal element of S is not a penetrator node.*

Proof. The proof process is similar to Lemma 4. □

4.3 Initiator Analysis for TBPKI-2 Protocol

4.3.1 Confidentiality Analysis of Initiator

Proposition 5. *Assuming the following conditions are valid:*

1) Σ is a TBPKI-2 space, C is a bundle of Σ, s is a Initiator strand. And $s \in Init[A, B, N_a, N_b, CVT_A, CVT_B, K_i, K_r]$, with $C - hight(s) = 3$;

2) $K_a \notin K_p$, and $K_b \notin K_p$;

3) $N_a \neq N_b$, and N_a is the only origin in Σ; $K_i \neq K_r$, and K_i is the only origin in Σ.

Therefore, for any node $m \in C$ *satisfying* $N_b \sqsubset term(n)$, $\{CVT_A N_a K_i Sign_a\} \sqsubset term(m)$ *is established or* $\{CVT_B N_a \{N_b\}_{K_{ab}} K_r Sign_b\} \sqsubset term(m)$ *is established. Specially,* $N_a \neq term(m)$.

The proof process is same as 4.2.2. It can obtain the confidentiality of N_a.

4.3.2 Consistency Analysis of Initiator

Proposition 6. *Assuming the following conditions are valid:*

1) Σ *is a TBPKI-2 space,* C *is a bundle of* Σ, s *is a responder strand. And* $s \in Init[A, B, N_a, N_b, CVT_A, CVT_B, K_i, K_r]$, *with* $C - hight(s) = 3$;

2) $K_a \notin K_p$, *and* $K_{ab} \notin K_p$;

3) $N_a \neq N_b$, *and* N_a *is the only origin in* Σ.

Therefore, C *contains a responder strand* $t \in Resp[A, B, N_a, N_b, CVT_A, CVT_B, K_i, K_r]$, *with* $C - hight(t) = 2$.

Proof. Here is a brief proof. Considering the set $\{m \in C : \{CVT_B, N_a, \{N_b\}_{K_{ab}}, K_r, Sign_b\} \sqsubset term(m)\}$ contains node $< s, 2 >$, so it is nonempty. And it has a minimal element m_0. If m_0 is on a regular strand t, then $t \in Resp[A, B, N_a, N_b, CVT_A, CVT_B, K_i, K_r]$. And t at least has two nodes in C.

If m_0 lies on a penetrator strand t, it can be proved that t is a penetrator strand of type E and its trace is $\{-K_b^{-1}, -CVT_B N_a \{N_b\}_{K_{ab}} K_r, +CVT_B N_a \{N_b\}_{K_{ab}} K_r Sign_b\}$. However, this contradicts Proposition 5, so N_a cannot appear on a node like $< t, 2 >$.

The conclusion on uniqueness corresponding to Proposition 3 can be proved similarly. □

4.4 Other Confidentiality Analysis of TBPKI-2 Protocol

The information that TBPKI-2 protocol needs to keep confidential also includes K_i and K_r. Because the first step $A \to B : CVT_A, N_a, K_i, Sign_a$ and the second step $B \to A : CVT_B, N_a, \{N_b\}_{K_{ab}}, K_r, Sign_b$ in the protocol sending process, K_i and K_r are not encrypted. Therefore, penetrator P can obtain $K_i(g^x)$ and $K_r(g^y)$. The x, y contained in them are secret data. So penetrator P can deduce the secret data x, y and send the secret data through the strand S. Therefore, K_i and K_r cannot guarantee the confidentiality.

5 Improvement

For the improvement of TBPKI-2 protocol, the information sent between A and B is encrypted after private key signature. As follows:

1) $A \to B : \{CVT_A, N_a, K_i, Sign_a\}_{K_b}$.
$Sign_a = \{CVT_A, N_a, K_i\}_{K_a^{-1}}$, indicates the signature of subject A for this message;

2) $B \to A : \{CVT_B, N_a, \{N_b\}_{K_{ab}}, K_r, Sign_b\}_{K_a}$.
$Sign_b = \{CVT_B, N_a, \{N_b\}_{K_{ab}}, K_r\}_{K_b^{-1}}$, indicates the signature of subject B for this message;

3) $A \to B : \{N_b - 1\}_{K_{ab}}$.

Because the private key of subject is unbreakable, the improved protocol can prevent Man-in-the-Middle Attack [14]. Therefore it can solve the hidden danger of possibly obtaining confidential information in 4.4 confidentiality analysis. After the improvement, the security of the protocol is guaranteed and the purpose of the protocol can be achieved. That is, secret key negotiation is performed while the identity of the communication subject is verified.

6 Comparison with Other Method

BAN logic pioneered the formal analysis of security protocols and has been widely appreciated for its simplicity and practicality. However, BAN logic can only analyze the authentication nature of the protocol to find its flaws, but cannot analyze the confidentiality nature of the protocol to ensure the security of the protocol. Compared with this method, strand space theory has the following advantages:

1) In the strand space model, the meaning of security protocol correctness includes both consistency and confidentiality. So the analysis scope of BAN logic is expanded;

2) The strand space model accurately describes the possible behaviors of penetrators in the system;

3) The strand space model is simpler to prove the correctness of security protocols and can more accurately confirm the assumptions made.

7 Conclusion

TBPKI-2 protocol can effectively prevent replay attacks, malicious tampering of information and other common attacks by ensuring the freshness of the temporary value and the unsolvability of the subject's private key. And it also can realize the purpose of confirming the source of information. However, it has the drawback of being intercepted by the penetrator and cracking the session key, so it cannot effectively achieve the purpose of key negotiation. Therefore, the TBPKI-2 protocol needs to be further improved. Because the private key of the subject is not cracked, it can be encrypted by public key before sent. It can prevent Man-in-the-Middle Attack during message transmission, and securing the security of the protocol.

Acknowledgments

This study was supported by the Natural Science Foundation of Anhui University (KJ2020A0034), the crosswise project entrusted by Anhui Yunxinfu Information Technology Co., Ltd. to Huaibei Normal University in 2021 (Construction of Information security technology guarantee system of information management platform). The authors gratefully acknowledge the anonymous reviewers for their valuable comments.

References

[1] Z. Chen, W. Liao, S. Shen, and H. Wang, "Wireless network authentication protocol based on PKI mechanism optimization," *Computer Engineering and Design*, vol. 33, no. 9, pp. 3297–3300, 2012.

[2] S. Chiou, H. Pan, E. F. Cahyadi, and M. Hwang, "Cryptanalysis of the mutual authentication and key agreement protocol with smart cards for wireless communications," *International Journal of Network Security*, vol. 21, no. 1, pp. 100–104, 2019.

[3] R. Focardi and F. L. Luccio, "Secure key management policies in strand spaces," in *Protocols, Strands, and Logic - Essays Dedicated to Joshua Guttman on the Occasion of his 66.66th Birthday*, Lecture Notes in Computer Science, D. Dougherty, J. Meseguer, S. A. Mödersheim, and P. D. Rowe, Eds., vol. 13066. Springer, pp. 175–197, 2021.

[4] Z. Guo, "Cryptanalysis of a certificateless conditional privacy-preserving authentication scheme for wireless body area networks," *International Journal of Electronics and Information Engineering*, vol. 11, no. 1, pp. 1–8, 2019.

[5] S. Hagihara, M. Shimakawa, and N. Yonezaki, "Verification of verifiability of voting protocols by strand space analysis," in *Proceedings of the 8th International Conference on Software and Computer Applications, ICSCA'19*, pp. 363–368, 2019.

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

[7] J. Liu, Y. Lai, S. Yang, and L. Xu, "Bilateral authentication protocol for WSN and certification by strand space model," vol. 46, no. 9, pp. 169–175, 2019.

[8] Y. Liu, Q. Meng, X. Liu, J. Wang, L. Zhang, and C. Tang, "Formal method for security analysis of electronic payment protocols," *IEICE Transactions on Information Systems*, vol. 101-D, no. 9, pp. 2291–2297, 2018.

[9] S. Pinsky, "Joshua guttman: Pioneering strand spaces," in *Protocols, Strands, and Logic - Essays Dedicated to Joshua Guttman on the Occasion of his 66.66th Birthday*, Lecture Notes in Computer Science, D. Dougherty, J. Meseguer, S. A. Mödersheim, and P. D. Rowe, Eds., vol. 13066. Springer, pp. 348–354, 2021.

[10] F. Journal of Thayer, J. C. Herzog, and J. D. Guttman, "Strand spaces: Why is a security protocol correct?" in *IEEE Symposium on Security and Privacy*, pp. 160–171, 1998.

[11] J. Yan, S. Ishibashi, Y. Goto, and J. Cheng, "A study on fine-grained security properties of cryptographic protocols for formal analysis method with reasoning," in *IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation, SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI 2018, Guangzhou, China, October 8-12, 2018*, G. Wang, Q. Han, M. Z. A. Bhuiyan, X. Ma, F. Loulergue, P. Li, M. Roveri, and L. Chen, Eds. IEEE, pp. 210–215, 2018.

[12] F. Yang, S. Escobar, C. A. Meadows, and J. Meseguer, "Strand spaces with choice via a process algebra semantics," *CoRR*, vol. abs/1904.09946, 2019. (http://arxiv.org/abs/1904.09946)

[13] M. Yao, J. Zhang, and X. Weng, "Research of formal analysis based on extended strand space theories," in *15th International Conference on Intelligent Computing Theories and Application (ICIC'19)*, Lecture Notes in Computer Science, vol. 11644, 2019.

[14] E. Ylli and J. Fejzaj, "Man in the middle: Attack and protection," in *Proceedings of the 4th International Conference on Recent Trends and Applications in Computer Science and Information Technology*, pp. 198–204. 2021.

[15] L. Yu, Y. Guo, and M. Jiang, "Improvement of strand space theory for application of minimal element method," *Quarterly Journal of Indian Pulp and Paper Technical Association*, vol. 30, no. 1, pp. 94–105, 2018.

[16] L. Yu, Y. Y. Guo, Z. P. Zhuo, and S. M. Wei, "Analysis and improvement of otway-rees based on enhanced authentication tests," *International Journal of Network Security*, vol. 23, no. 3, pp. 426–435, 2021.

[17] L. Yu, Z. Y. Yang, and Z. P. Zhuo, "Extension of pcl theory and its application in improved ccitt x.509 analysis," *International Journal of Network Security*, vol. 23, no. 2, pp. 305–313, 2021.

Biography

Wen-bei Zong was born in 2000. She received the MS degree in software engineering from Huainan Normal University of China. Currently, She is a graduate student in the school of computer science and technology, Huaibei Normal University, China. Her research interests include cryptography and information security. (12111080780@chnu.edu.cn)

Lei Yu was born in 1978. He received the MS an BS

degree in computer science and technology from Huaibei Normal University of China. Currently, he is an assistant professor and MS supervisor in the school of computer science and technology, Huaibei Normal University, China. His major research interests include cryptography and information security. He has published many papers in related journals.(yulei@chnu.edu.cn)

Research on the Infringement of Personal Information by Web Crawlers Based on Legal Regulation

Qingyuan Liu[1] and Feng'e Huo[2]
(Corresponding author: Feng'e Huo)
Office of Academic Affairs, Cangzhou Normal University[1]
School of Politics and History, Cangzhou Normal University[2]
Cangzhou, Hebei 061001, China
Email: e944070@163.com
(Received Nov. 30, 2018; Revised and Accepted Oct. 11, 2022; First Online Oct. 15, 2022)

Abstract

Web crawlers have been widely used to collect data and information efficiently. This paper first analyzed the application of web crawlers in information crawling, introduced the topic crawler and HITS algorithm, and proposed an improved HITS algorithm to improve the crawling accuracy of web pages. The experimental analysis verified that the improved algorithm was reliable in enhancing the crawling accuracy and always had a crawling accuracy above 0.7 when crawling different topics, higher than the traditional HIS algorithm. Then, this paper analyzed the infringement of personal information by web crawlers, described some relevant cases, pointed out the shortcomings of existing laws, and gave some suggestions. This work supports further improving the rule system needed for crawler applications and punishing personal information infringement.

Keywords: Legal Regulation; Personal Information; Privacy Protection; Web Crawlers

1 Introduction

With the rapid development of technology, information has become an important resource that plays a huge role in many fields [9]; therefore, access to information has become an increasingly essential issue. Web crawlers refer to a technology that crawls specific data according to preset algorithms and procedures [12], which is a good way to obtain information [3]. With the progress of the Internet, web data is increasing, and web crawlers can help users to crawl the information needed from a massive number of pages [13].

At present, web crawler technology is in continuous development and has been well-used in many fields [1]. Luo *et al.* [4] crawled the data of Baijiu in the Tmall platform based on web crawler technology and analyzed the impact of the seller's credit, brand, and service on the sales of goods. Yu *et al.* [11] analyzed the web crawler technology and applied the PageRank algorithm in a topic crawler to build a vertical search engine. Surahman *et al.* [8] used a web crawler to subdivide the online products of e-marketplaces, including Tokopedia, Shopee, and Bukalapak, and obtained a success rate of 79%.

Pramudita *et al.* [7] developed a multi-threaded web crawler application and also adjusted the web structure to make it crawl web information more efficiently. However, while sharing and utilizing data and information, the problems of information leakage, data misuse, and privacy violation brought about by web crawlers [6] have provoked thought. Individual users cannot enjoy the value of big data but must face the risk of privacy infringement. This paper analyzed the application of web crawlers in information crawling, explained the violation of personal information by web crawlers through specific cases, analyzed the illegality of web crawlers from the perspective of laws, and proposed some suggestions based on the shortcomings of existing laws. The present study provides a theoretical basis for the reasonable and legal use of web crawlers in practice.

2 Application of Web Crawlers in Information Crawling

In the era of big data, the collection, arrangement, and application of web data are beneficial to improve the efficiency of obtaining information and can help various enterprises to catch the market trends and grasp the user behavior to create greater benefits [10]. The process of crawling information with web crawlers is as follows. Crawlers access through initial Uniform Resource Locators (URLs) to download and save the content, parse

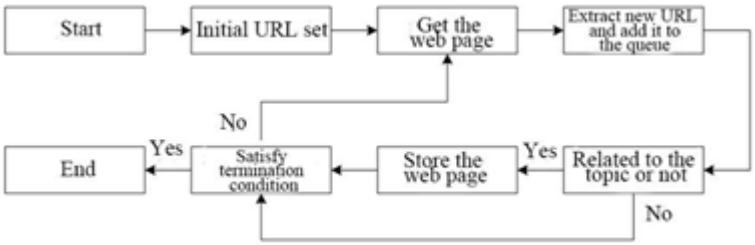


Figure 1: The workflow of a topic crawler

links to other pages from the content, and crawl them until reaching a predetermined running time or a predetermined number of pages. A topic crawler is a web crawler that aims to crawl content related to a specified topic, which can effectively save broadband resources. The workflow of a topic crawler is shown in Figure 1.

According to Figure 1, the topic crawler discards the pages that do not match the topic and then stores the pages that match the topic when crawling the web content, and this process is repeated over and over until it reaches the criteria set by the system. The HITS algorithm [2] is an important algorithm for crawling pages in the topic crawler, which assigns two attributes to every page: hub and authority. The authority page refers to the page that best matches the keyword, and the hub page is the authority page where multiple links exist. When crawling pages, the HITS algorithm can sort the pages related to the topic according to the authority value from highest to lowest. The principle of the HITS algorithm is to determine the web subgraph $G = (V, E)$ related to the topic and calculate the hub and authority values for every page. It is assumed that there are n nodes in G. The authority and hub values of node i are written as $a_i(v)$ and $h_i(v)$. The specific calculation formulas are:

$$\begin{aligned} a_i(v) &= \sum_{(w,v)\in E} h_{i-1}(w), \\ h_i(v) &= \sum_{(v,w)\in E} a_{i-1}(w). \end{aligned}$$

To ensure the invariance of the results, after every calculation, $a_i(v)$ and $h_i(v)$ are normalized, and the formulas are written as:

$$\begin{aligned} a_i(v) &= \frac{a_i(v)}{\sqrt{\sum_{q\in n}[a(q)]^2}}, \\ h_i(v) &= \frac{h_i(v)}{\sqrt{\sum_{q\in n}[h(q)]^2}}. \end{aligned}$$

In web analysis, the HITS algorithm still has some shortcomings. First, it tends to ignore new pages, which is because the newly appeared pages are not easily found by the HITS algorithm due to fewer links to other pages; second, the HITS algorithm also tends to cause the problem of topic drift: if there are some pages that point to a lot but are not strongly related to the topic, the HITS algorithm also tends to give them high authority values. This paper proposed an improved HIT algorithm based on weight value $p(f)$, which is a value related to the modification time of a web page and the number of comments on the web page. For a web page, if there is a large difference between the query time and the time of its last modification, it means that the reference value of this web page is small; if the number of its comments (more than ten words as a valid comment) decreases, it means that the value of the web page decreases, which can be described by:

$$f(t,k) = \begin{cases} t, & m < 10 \\ \frac{t}{\log_k m}, & m \geq 10, k > 0 \end{cases}$$

where t refers to the difference between the time of crawling a page and the last modified time of the page (unit: month), m refers to the number of comments on the page, and k is the log bottom number. Ultimately, weight value $p(f)$ is calculated by:

$$p(f) = \begin{cases} 1, & t \leq 1 \\ \frac{t}{\ln(f)}, & t > 1 \end{cases}$$

Based on $p(f)$, the calculation formulas of $a_i(v)$ and $h_i(v)$ of the improved HITS algorithm are updated as:

$$\begin{aligned} a_i(v) &= \sum_{(w,v)\in E} h_{i-1}(w) \times p(f), \\ h_i(v) &= \sum_{(v,w)\in E} a_{i-1}(w) \times p(f). \end{aligned}$$

To understand the performance of the improved HITS algorithm, this paper conducted an experiment on Windows 10 operating system. Let the number of topic-related pages crawled be N and the number of all web pages crawled be T. The performance of the topic crawler was evaluated in terms of the crawling accuracy rate, which was calculated by:

$$Accuracy = \frac{N}{T}.$$

Relevant web pages were crawled taking education as the topic. The change in the crawling accuracy of the HITS and improved HITS algorithms with the increase of crawled pages is shown in Figure 2.

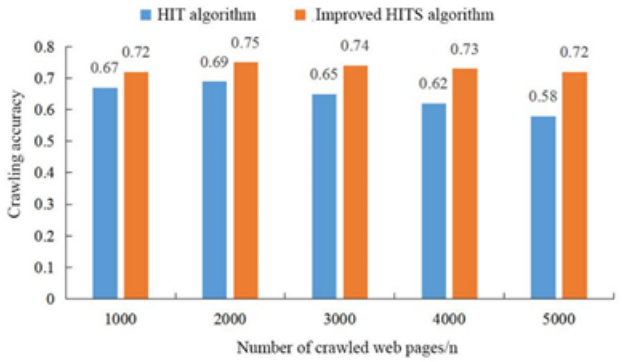


Figure 2: Theme crawler performance comparison

It was seen from Figure 2 that when the number of crawled pages was 1000, the crawling accuracy of the improved HITS algorithm was 0.05 higher than that of the HITS algorithm (0.72 vs. 0.67); when the number of crawled pages reached 2000, the crawling accuracy of the two algorithms reached the highest value, 0.69 and 0.75 respectively, and the crawling accuracy of the improved HITS algorithm was 0.06 higher than that of the HITS algorithm. Then, as the number of crawled pages increased, the number of topic-independent pages crawled by the algorithm also increased, leading to a decrease in the crawling accuracy. When the number of crawled pages reached 5000, the crawl accuracy of the HITS algorithm dropped to 0.58, and the crawl accuracy of the improved HITS algorithm dropped to 0.72, but it was still 0.14 higher than that of the HITS algorithm. The experimental results demonstrate the reliability of the improved HITS algorithm in crawling web content. The improved HIS algorithm could ensure good crawling accuracy even when the number of web pages crawled was large.

To further understand the performance of the improved HITS algorithm, the performance of the two algorithms was compared when crawling different topics, and the results are shown in Figure 3.

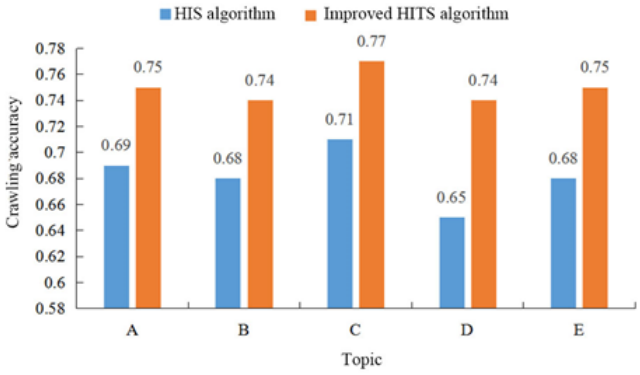


Figure 3: Comparison of the crawling accuracy between algorithms on different topics

It was noticed from Figure 3 that the crawling accuracy of the improved HITS algorithm was always higher than that of the HITS algorithm in crawling different topics. The maximum and minimum values of the crawling accuracy of the HITS algorithm were 0.71 and 0.65, while the crawling accuracy of the improved HITS algorithm was always above 0.7, with a minimum value of 0.74 and a maximum value of 0.77. These results further proved the excellent performance of the improved HITS algorithm in crawling web pages.

3 Violation of Personal Information by Web Crawlers

According to the previous section, it is found that web crawlers have significant advantages in crawling information; however, as crawlers are more and more widely used, there are also many abuse cases of crawlers. The improper use of web crawlers not only consumes computer resources excessively [5] but also leads to the leakage of information, which also includes personal information.

Personal information contains many human behavior data, such as personal preferences, occupation, health, etc. It is also closely related to national security, and the illegal crawling and use of personal information by web crawlers is an act of infringement. The infringement acts of web crawlers on individual users include:

1) Creating crawler software and selling it to others for profit;

2) Creating crawler software to crawl personal information;

3) Purchasing the right to use crawler software and crawling personal information;

4) Purchasing the right to use crawler software to crawl personal information and selling it for profit;

5) Serving on a platform company that uses crawlers to obtain user information and obtain personal information to sell for profit.

The number of cases involving web crawlers from 2013 to 2020 is shown in Table 1.

It was seen from Table 1 that the number of cases concerning web crawlers had seen a significant increase in recent years, increasing from 15 cases in 2019 to 27 cases in 2020. The distribution of applicable charges is shown in Table 2.

Some examples of cases related to violating citizens' personal information are as follows. In 2016, Weimeng company, the operating entity of Weibo, sued TalkU company (the operating entity of Maimai software). During the cooperation period, Weimeng company inferred that the defendant had illegally captured information about Weibo users through crawler technology and that Maimai still used a large amount of basic information about Weibo

Table 1: Number of cases related to web crawlers from 2013 to 2020

	2013	2014	2015	2016	2017	2018	2019	2020
Number of cases	1	2	1	1	4	8	15	27

Table 2: Applicable charges in relevant cases

Applicable charges	Number of cases
The crime of infringing on citizens' personal information	23
The crime of illegally obtaining computer information system data	10
Copyright infringement crime	8
The crime of spreading obscene materials for profit	4
The crime of providing programs and tools for intruding and illegally controlling computer systems	3
Others	11

users in the months after the cooperation between the two parties ended in August 2014. The court found that the parties had a competitive relationship and constituted unfair competition. From 2019 to 2020, without the permission of Company E, Company Z illegally obtained the order information, merchant information, and other data of Company E through a crawler program, and Company Z was prosecuted for the crime of illegally obtaining computer information system data.

In April-May 2020, the suspects Chen and Wu used crawler programs and other means to illegally obtain more than 90,000 personal information, including names, ID cards, cell phone numbers, etc. They were suspected of violating personal information and were sentenced to fixed-term imprisonment and a fine.

In 2021, a Shanghai information technology company was prosecuted for infringing on citizens' personal information by crawling personal information such as ID card, social security, provident fund, credit card, and communication records more than 3.08 million pieces through crawling technology without obtaining approval from the relevant state departments and obtaining illegal gains more than 17.5 million yuan.

4 Legal Regulation of Web Crawlers

Legal regulation is the main regulatory method for the infringement of personal information by web crawlers. However, at present, there is no clear domestic regulation for the infringement of personal information by web crawlers.

For the protection of personal information, the content about whether it is illegal for crawlers to crawl personal information is scattered in many laws:

1) The content about the principles of handling personal information in the Civil Code;

2) The limits of operators in crawling and using personal information through crawler technology in the Law of the People's Republic of China Against Unfair Competition;

3) The content about data security protection obligations in the Data Security Law of the People's Republic of China.

In the current legal regulation, there are also some issues that need to be addressed.

1) The interpretation of the provision of "violation of relevant state regulations" in the "crime of infringing citizens' personal information" is unclear.

2) There is an expansion of judgment on the object of the act of infringement, which deviates from the original purpose of protecting the legal interests of the crime of infringement of citizens' personal information.

3) The criteria for judging "illegal access" are rough and vague, making it difficult to deal with the increasingly complex illegal acts of web crawlers.

4) There is a lack of Detailed and specific regulations for the determination of the infringement results of the act, which cannot meet the requirements of reality.

The legal regulation of web crawlers is conducive to protecting personal information but restricts the development of network technology. In order to balance the protection of personal information and the flow of data value, the following suggestions are proposed for the current legal regulation:

1) Further clarifying the specific meaning of "violation of relevant state regulations";

2) Further clarifying the scope of citizens' personal information in the criminal law, and personal information that has undergone multiple transformations or needs to be combined with multiple other information before identification should not be included in the scope of protection;

3) Further regulating the access rights of crawler technology and clarify the criteria for judging the act of "illegal access": the act of publicly crawling data under the premise of following the robots agreement does not constitute an illegal act, but the act of forcibly crawling others' data in violation of the robots agreement may constitute unfair competition;

4) Further clarifying the infringement of the legal interests of crawling personal information with crawlers and determining which behaviors need to be regulated and punished by criminal law.

5 Conclusion

This paper studied web crawlers. First, an improved HITS algorithm was designed, and a high crawl accuracy of the algorithm on web crawling was obtained through experimental analyses, which proved the reliability of the method for information crawling. Then, the infringement of web crawlers on personal information was analyzed, and several relevant cases were introduced. Finally, the current legal regulation shortage was pointed out, and some suggestions were given for balancing personal privacy protection and encouraging technological innovation.

References

[1] T. Z. Ahram, D. Nicholson, "Using Dark Web Crawler to Uncover Suspicious and Malicious Websites," in *Advances in Intelligent Systems and Computing*, vol. 782, pp. 108-115, 2019.

[2] G. A. Al-Sultany, I. K. Abbood, "Conferences events suggestion using ranked based hyperlink-induced topic search algorithm," *Journal of Computational and Theoretical Nanoscience*, vol. 16, no. 3, pp. 982-988, 2019.

[3] N. Hien, T. Q. Tien, V. H. Nguyen, "Web crawler: Design and implementation for extracting article-like contents," *Cybernetics and Physics*, vol. 9, no. 3, pp. 144-151, 2020.

[4] M. Luo, J. Lin, "Research on the impact of credit, brand and service recovery on online sales based on web crawler technology and regression analysis methods," *Journal of Physics Conference Series*, vol. 1955, no. 1, pp. 1-8, 2021.

[5] R. Nath, N. Kumar, S. Tuteja, "A survey on reduction of load on the network," *Advances in Intelligent Systems and Computing*, vol. 321, pp. 239-249, 2015.

[6] S. Pastrana, D. R. Thomas, A. Hutchings, R. Clayton, "CrimeBB: Enabling cybercrime research on underground forums at scale," in *World Wide Web Conference*, vol. 2018, pp. 1845-1854, 2018.

[7] Y. D. Pramudita, D. R. Anamisa, S. S. Putro, M. A. Rahmawanto, "Extraction system web content sports new based on web crawler multi thread," *Journal of Physics: Conference Series*, vol. 1569, no. 2, pp. 1-6, 2020.

[8] A. Surahman, A. F. Octaviansyah, D. Darwis, "Ekstraksi data produk e-marketplace sebagai strategi pengolahan segmentasi pasar menggunakan web crawler," *Sistemasi: Jurnal Sistem Informasi*, vol. 9, no. 1, pp. 73, 2020.

[9] N. Tempini, "Till data do us part: Understanding data-based value creation in data-intensive infrastructures," *Information and Organization*, vol. 27, no. 4, pp. 191-210, 2017.

[10] Y. Wang, Q. Zhong, "Data mining of popular science books based on web crawler," in *International Conference on Big Data, Artificial Intelligence and Internet of Things Engineering (ICBAIE'20)*, pp. 92-95, 2020.

[11] L. Yu, Y. Li, Q. Zeng, "Design of topic web crawler based on improved pagerank algorithm," *Journal of Physics: Conference Series*, vol. 1754, no. 1, pp. 1-7, 2021.

[12] X. Zhang, Z. Cheng, C. Zhang, "Design and implementation of a web crawler system based on an adaptive page-rank algorithm," *Journal of Physics: Conference Series*, vol.1634, pp. 012021, 2020.

[13] J. Zhou, S. Wang, L. Li, "Application of Python web crawler technology in infodemiology," *Chinese Journal of Epidemiology*, vol. 41, no. 6, pp. 952-956, 2020.

Biography

Qingyuan Liu, born in November 1991 in Cangzhou, Hebei Province, has received the master's degree. She is a lecturer. Her main research direction is civil and commercial law.

Feng'e Huo, born in September 1978 in Haixing, Hebei Province, has received the master's degree. She is an associate professor. Her main research direction is intellectual property law.

Revocable Outsourced Decryption of CP-ABE Based on OBDD

Li Chen[1], Rui Guo[1], Lei Xu[1], Xin Wei[1], Geng Yang[1], Chaoyuan Zhuang[1], and Qianqian Zhao[2]
(Corresponding author: Li Chen)
School of Cyberspace Security, National Engineering Laboratory for Wireless Security,
Xi'an University of Posts and Telecommunications[1]
Xi'an 710121, China.
Shanghai Research and Development Center for Micro-Nano Electronics, Shanghai, China[2]
Email: chenli20225@163.com
(Received May 13, 2022; Revised and Accepted Oct. 9, 2022; First Online Oct. 15, 2022)

Abstract

Ciphertext-policy attribute-based encryption (CP-ABE) has proven to be an effective method for sharing data on cloud servers. CP-ABE schemes have traditionally used access structures like LSSS, access trees, threshold gates, and AND gates. Meanwhile, they have a heavy computational overhead on users. This paper presents a CP-ABE scheme based on an ordered binary decision diagram (OBDD) with revocable outsourcing decryption. A significant advantage of OBDD is its extensive expressiveness. It supports logical operations, such as "AND", "OR", and "NOT", as well as positive and negative attributes. In the OBDD, the complexity of the encryption algorithm relates to the number of valid paths rather than the number of attributes appearing in the access policy. The scheme enables the prevention of user collusion when the revocation of an attribute occurs. We use an attribute group that assigns a unique value to each user. To reduce the computational load on users, our scheme outsources the decryption task to the decryption server while favoring the verification of the converted ciphertext. Furthermore, we demonstrate that the proposed scheme is secure under the CDH hardness assumption. Finally, we compared our scheme to other communication and overhead computation schemes.

Keywords: Attribute Revocation; OBDD; Outsourced Computation

1 Introduction

Cloud storage is an essential service of cloud computing, which offers excellent convenience for data owners to share their data. Users would prefer to save their private data on cloud services for easy access anytime and anywhere. Enterprises tend to store their files on cloud services for collaboration among employees. Although cloud services bring great convenience to people, data security, including data leakage and illegal access, has become a significant obstacle to its popularity. Some researchers have provided some solutions such as data access control [7,17] encrypted cloud data search services [9,16], attribute-based encryption (ABE) [1,14], and so on. Among these solutions, ABE is one of the most effective solutions due to its one-to-many encryption feature. ABE is classified into ciphertext-policy attribute-based encryption (CP-ABE) and key-policy attribute-based encryption (KP-ABE). In CP-ABE, the policy is contained in the ciphertext, and the user's key is associated with the attribute. In KP-ABE, the opposite is true. CP-ABE is better suited for cloud services.

However, the existing CP-ABE schemes are based on the common LSSS [29], access trees [2], threshold gates [22], and AND gates [28]. There is very little research work on CP-ABE based on the OBDD. The OBDD has the advantage of great flexibility and expressiveness. In this work, the most intractable problem at present is attribute revocation. An attribute could be owned by several users, and a user also may have several attributes, so a user's attribute revocation will impact other users with the same attribute. [8] introduced the idea of attribute groups. In their scheme, the same attribute group keys are assigned to the same attributes. When an attribute revocation occurs, the attribute manager (AM) renews the attribute group keys for the existing users. However, the scheme suffers from the collusion attacks of existing users and revoked users. While Li *et al.* [11] have solved the issue of collusion attacks, the scheme requires users to have high computational power.

CP-ABE schemes generally exhibit the drawback that the ciphertext length and decryption cost increase with the complexity of the access policy. It imposes a heavy computational burden on the user. Despite there being some ABE schemes with a fixed ciphertext size, their access policies are limited to AND gates [3] or threshold

gates [22]. [23] provided an outsourced decryption solution in CP-ABE. The complicated decryption overhead is undertaken first by the decryption server (DS). The user then performs simple calculations. Although this way greatly reduces the computational overhead for the user, it lacks the verification of the correctness of the ciphertext. Afterward, methods with verifiable outsourced decryption were adopted by many scholars [6,21]. Considering the above issues, we will attempt to address the problem of revocable outsourced decryption based on OBDD in CP-ABE.

1.1 Related Work

Several researchers have contributed to overcoming the above challenges. In this part, we analyze the previous endeavors and the relevant literature.

1) Access Structure in CP-ABE
Access structures play a crucial role in CP-ABE. It has been studied by many researchers, including AND gates, threshold gates, LSSS, and access trees. AND gates can be subdivided into AND gate supporting a single positive value without wildcards [22], AND gate supporting positive and negative values without wildcards [4], AND gate supporting positive and negative values with wildcards [20], AND gate supporting multiple values without wildcards [19] and AND gate supporting multiple values with wildcards [26], where AND gate supporting multiple values with wildcards are the most flexible. But it only supports a type of AND logical operator. Access trees are an extension of thresholds. It supports not only thresholds, but also "AND" and "OR" logical operations. However, it needs to compute the secret value stored in the root node. LSSS is preferred by many researchers, because of its fine-grained expression. But a secret sharing value must be specified in LSSS. The OBDD access structure proposed by Li *et al.* [12] was first applied to CP-ABE. OBDD supports not only positive and negative attributes but also "AND", "OR", and "NOT" logical operators. The most important feature is that the complexity of the encryption algorithm is related to the number of valid paths, not the number of attributes appearing in the access structure. However, the attribute revocation is not addressed in [12]. Edemacu *et al.* [5] presented an attribute revocation method based on OBDD. Unfortunately, users' computation overhead is still large.

2) Revocation in CP-ABE
A flexible and direct user revocation of CP-ABE was introduced in [25]. It has a fixed size of ciphertext. In addition, only part of the ciphertext is updated when a revocation event occurs. However, this way does not support attribute revocation. Besides, the data owner must maintain the revocation list. [15] designed a CP-ABE scheme with a time-based direct revocable feature. They adopted a secret key time validation technique. Although this method effectively reduces the revocation list maintained by the data owner, it relies on the hierarchical identity-based encryption (HIE) mechanism. A CP-ABE of attribute revocation was presented in [8]. They introduced a new concept claimed as attribute group. It means that only a group of users with the same attribute has an attribute group key that only they know. But there is a conspiracy attack. Our paper uses the idea of [11] to build a new attribute revocation scheme for CP-ABE based on OBDD. It also is resistant to user collaboration attacks.

3) Outsourced computation in CP-ABE
CP-ABE schemes need a lot of exponential operations and bilinear mappings during encryption and decryption, which results in a high computing cost. Outsourcing techniques are proposed in CP-ABE [27], which design a key blind technique without revealing the key. The private key of a user is composed of a "retrieval key" (RK) and a transformation key (TK). The RK is only accessible by the user. The TK was sent to the decryption server (DS). With the assistance of the TK, the DS decrypts the conversion ciphertext. The user can use the RK to decrypt the message after only one exponentiation operation. Subsequently, considerable works have focused on outsourcing algorithms in different phases of CP-ABE, such as the private key generation [24], encryption [13], and decryption [10]. However, there is a problem that the user cannot confirm the DS performs the computation outsourcing honestly. [18] described verifiable outsourced decryption in CP-ABE. Unfortunately, [18] needed an extra verification key for validation. In our work, we do not.

1.2 Our Contributions

The paper provides a CP-ABE scheme using OBDD with revocable outsourced decryption. We overcame the problems of attribute revocation. Specifically, combining the idea of attribute groups, we assign a random value to each user, which is bound to the attribute group key. The OBDD is flexible and expressive, supporting positive and negative values of the attribute as well as "OR", "AND", and "NOT" logical operators. Moreover, users have a low computation overhead. The main contributions of this study can be summarized as follows.

1) We solve the attribute revocation in CP-ABE based on OBDD, which is resistant to user collusion.

2) To relieve the computational burden on the user, we adopt a verifiable outsourcing decryption approach. Especially, without the need for extra verification keys, the user enables to verify the validity of the converted ciphertext.

Table 1: Description of the parameters

Notations	Meanings
$OBDD$	Access structure.
Γ	Set of user attributes.
O	Attribute sets in the access structure.
M	Message.
PK	System public key .
MSK	System private key.
SK	Private key of the user.
CT	Ciphertext.
$\hat{CT}$	Re-encrypted ciphertext.
CT^*	Ciphertext after attribute revocation.
KEK_i	Attribute group key for attribute i.
KEK_i^*	Attribute group key after revoking attribute i.
Hdr	A header message.
Hdr^*	A header message after attribute revocation.
TT	Transformation ciphertext.
TK	Transformation Key.
RK	Retrieving key.

3) Similar schemes were compared with ours in terms of functionality, computational and communication costs. This result shows that ours has practical applications and lower computing and communication costs.

1.3 Organization

The following is the structure of the paper. Section 2 contains the preliminaries. Section 3 describes a specific ABE scheme. Section 4 demonstrates the safety of our scheme. We analyze our scheme's performance in Section 5. The paper is concluded in Section 6.

2 Preliminaries

2.1 Bilinear Maps

Suppose that there are two multiplicative cyclic groups G and G_T with the prime order p, and a generator g belonged to G. A bilinear map $\hat{e} : G \times G \rightarrow G_T$ meets the following characteristics:

1) *Bilinearity*: $\hat{e}(g_1^{z_1}, g_2^{z_2}) = \hat{e}(g_1, g_2)^{z_1 z_2}$, $g_1, g_2 \in G$ and $z_1, z_2 \in Z_p$.

2) *Non-degeneracy*: $\hat{e}(g_1, g_2) \neq 1$

3) *Computability*: $\forall g_1, g_2 \in G$, $\hat{e}(g_1, g_2)$ can be efficiently computed.

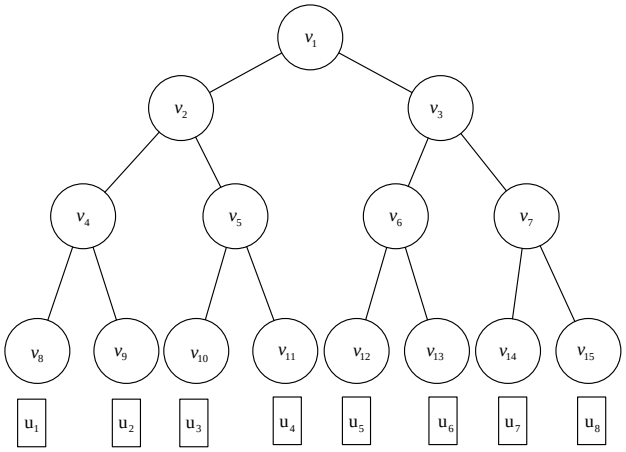


Figure 1: A KEK tree

2.2 CDH Assumption

CDH Problem: Given the triple $(g, g^{z_1}, g^{z_2}) \in G$, the CDH problem means that it calculates the value $g^{z_1 z_2}$ in G, where $z_1, z_2 \in Z_p^*$.

CDH Assumption: The CDH assumption holds if all probabilistic polynomial time adversaries $\mathcal{A}$ have a negligible advantage in solving the CDH problem. The advantage of $\mathcal{A}$ is defined as

$$Adv_{CDH}(\mathcal{A}) = |\Pr[\mathcal{A}(g^{z_1}, g^{z_2}) = g^{z_1 z_2}]| = \epsilon$$

2.3 OBDD Access Structure

A binary decision diagram (BDD) is a directed acyclic graph with root nodes. It can be represented by the Boolean function $f(x_1, x_2, x_3, ..., x_n)$ and has the following characteristics.

1) There is a unique root node.

2) All non-terminal nodes are expressed by Boolean variables.

3) Each non-terminal nodes have only two output edges, which are called 1 and 0 branches, respectively.

4) The terminal node can only be 0 or 1, and there is no child node.

5) In the path between the root node and the terminal node, each variable appears only once.

An OBDD is a BDD with a consistent ordering of variables on any path. In other words, all variables appear in the order of π (we set the order of π to be $x_1 < x_2 < x_3 < ... < x_n$) from the root node to the terminal node. The OBDD has quite significant advantages over BDD. It can not only express the Boolean function but also has the uniqueness of the Boolean function. Hence, in this paper, our focus is on OBDD.

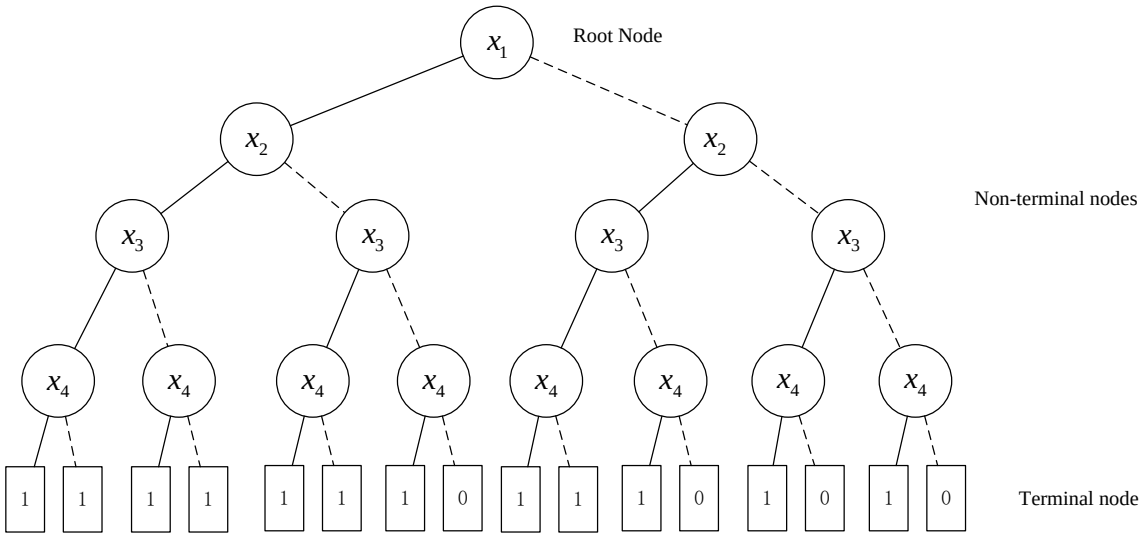


Figure 2: Original OBDD representation of $f(x_1, x_2, x_3, x_4) = x_1x_2 + x_1x_3 + x_2x_3 + x_4$

First, we must solve how to represent the OBDD with Boolean functions. The attribute variables in the attribute set O of the access policy are denoted by x_i $(1 \leq i \leq n)$. The attribute number is set to i $(1 \leq i \leq n)$, where n denotes the total number of attributes. In OBDD, all nodes except the terminal nodes are assigned unique node numbers id $(2 \leq i \leq n)$. Two terminal nodes are assigned 0 and 1, respectively. The expression form of OBDD is $\{Node(id, i, high(id), low(id)) \mid id \in ID,\ i \in I\}$. id denotes the node serial number. ID means the set of node serial numbers. i refers to the attribute number. I describes the set of attribute numbers. $high(id)$ denotes the node serial number of the child node of branch 1 connected to the node id. $low(id)$ denotes the node serial number of the child node of branch 0 connected to the node id.

Secondly, to ensure the user's attribute set Γ meets the OBDD access policy, the process is as follows:

1) The system consists of n attributes, which are represented by i. When the value of attribute i is a positive attribute value, it means that attribute i is included in the attribute set O of the access policy. On the contrary, attribute i is not in the attribute set O.

2) Starting from the root, the root node is set to the current node. In the next step, we compare the attributes represented by the current node with the elements in the attribute set O of the access policy. The current node is set to $high(id)$ if it exists; if not, $low(id)$ is set to the current node. The above process is repeated until the current node reaches the terminal node. If it reaches terminal node 1, the attribute set Γ of the user meets the OBDD access structure. Conversely, it is not satisfied.

At last, we provide a case. The access policy is (x_1 AND x_2) OR (x_1 AND x_3) OR (x_2 AND x_3) OR x_4. Its Boolean function is expressed as $f(x_1, x_2, x_3, x_4) = x_1x_2 + x_1x_3 + x_2x_3 + x_4$, where $\cdot, +$, which denotes "AND", "OR" logical operations, respectively. The square represents a terminal node, and its value can only be a Boolean constant 0 or 1, as shown in Figure 2. The circle represents a non-terminal node. It is connected to a solid line and a dashed line, respectively. The solid line indicates a node with the Boolean variable value of 1. The dashed line indicates a node with the Boolean variable value of 0. In Table 2, we assign unique node serial numbers and attribute numbers to all nodes.

Suppose a user A has a set of attributes (x_1, x_2, x_3, x_4) and user B has a set of attributes (x_1). Next, we judge whether user A and B meet the access policy. The path of user A is $x_1 \rightarrow x_2 \rightarrow x_3 \rightarrow x_4 \rightarrow 1$ and the path of user B is $x_1 \rightarrow x_2 \dashrightarrow x_3 \dashrightarrow x_4 \dashrightarrow 0$. The path from the root node to the terminal node with 1 is called a valid path, which means that the user satisfies the policy. It is clear that user A satisfies the access policy and user B does not. We can see a total of 6 valid paths in Figure 2. The paths was noted as $Path = \{Path_1, Path_2, Path_3, Path_4, Path_5, Path_6\}$, where $Path_1 = \{x_1, x_2, x_3, x_4\}$,$Path_2 = \{x_4\}$. However, we can also find that more than one terminal node with 1 or 0, which is a waste of storage costs. Hence, we reduce the duplicate nodes, as shown in Figure 3.

2.4 KEK Tree

The key-encryption key (KEK) tree is a complete binary tree based on the users' set. The system's user set is marked by $U = \{u_1, u_2, ..., u_n\}$. The attribute set is denoted by $X = \{x_1, x_2, ..., x_n\}$. AG_i is called the attribute group i, representing the set of users with attribute x_i. TA assigns group keys to all users in attribute group AG_i. If the attribute i of u_1 is revoked, the TA will redistribute the attribute group key to others in the attribute group

Table 2: Comparison of the functional features

Scheme	Expressiveness	Revocation	Collusion resistance	Outsourcing decryption	Verification
[8]	Access Tree	√	×	×	×
[11]	Access Tree	√	√	×	×
[12]	OBDD	×	×	×	×
[5]	OBDD	√	√	×	×
Ours	OBDD	√	√	√	√

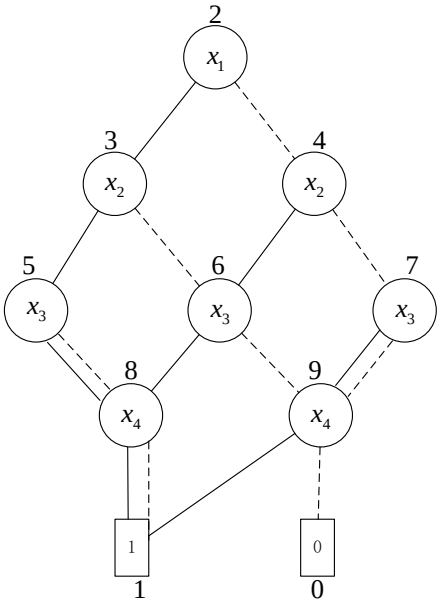


Figure 3: Improved OBDD representation of $f(x_1, x_2, x_3, x_4) = x_1x_2 + x_1x_3 + x_2x_3 + x_4$

AG_i except for u_1. The original attribute group key of u_1 is invalidated.

The detailed process for TA to build the KEK tree and generate key parameters related to the user is as follows:

1) In U, all users are assigned to the leaf nodes. TA stores a random value for each non-leaf node v_j.

2) Each user u_k has a set of path nodes, which contains all nodes from the terminal node to the root node.

3) Every attribute group has a minimal set that can cover all users in that attribute group.

4) Find the intersection of the user's set of path nodes with the minimum coverage set. The result is only one node v_j storing a random value θ_j.

For example, u_1 first calculates the set of path nodes, $path(u_1) = \{v_8, v_4, v_2, v_1\}$. We assume that the attribute group is $AG_1 = \{u_1, u_2\}$. TA computes $Min(AG_1) = \{v_4\}$ and the intersection of $Min(AG_1)$ with $path(u_1)$, $Min(AG_1) \cap path(u_1) = v_4$. If the attribute i of u_1 is revoked, the TA updates attribute group $AG_1 = \{u_2\}$ and works out the minimum coverage set $Min(AG_1) = v_8$.

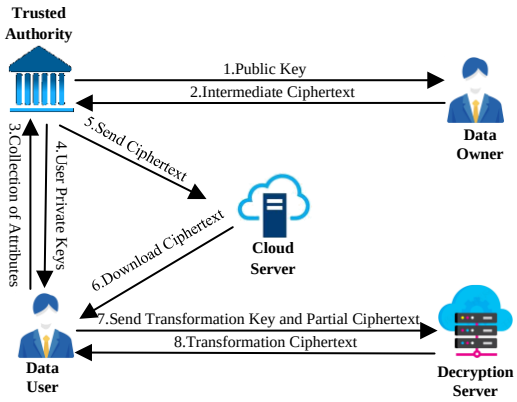


Figure 4: The system architecture of our scheme

At this time, the θ_2 originally distributed by TA will be invalid.

2.5 Security Model

The revoked users can conspire with existing users, resulting in unauthorized users accessing data. An adversary $\mathcal{A}$ can request two kinds of keys. Type-A is a user key that meets the challenge access structure O^* but the challenge attribute att_f^* is revoked. Type-B is a user key that does not meet the challenge access structure O^* but the user has the challenge attribute att_f^*. The following is the game between $\mathcal{A}$ and simulator $\mathcal{B}$:

Init: $\mathcal{A}$ chooses the access policy O^* of the challenge and the attribute att_f^* of the challenge to send to $\mathcal{B}$. The attribute att_f^* is critical for satisfying O^*.

Setup: The $\mathcal{B}$ first executes $Setup(1^\theta)$ to compute the public key PK and the master key MSK. Then, $\mathcal{B}$ updates the private and public keys corresponding to att_f^*. $\mathcal{B}$ gives PK to $\mathcal{A}$ and keeps MSK secret, ultimately.

Phase 1: $\mathcal{A}$ queries for two types of keys, which we label as Type-A and Type-B.

Type-A: u_1's attribute set Γ_{u_1} satisfies O^*, but the user's attribute att_f^* has been revoked. $\mathcal{B}$ runs the algorithm $TAKeyGen(PK, MSK, \Gamma_{u_1})$ to generate u_1's keys (DSK_{u_1}, KEK_{u_1}) and returns to $\mathcal{A}$.

Type-B: u_2's attribute set Γ_{u_2} does not satisfy O^*, but u_2 has attribute att_f^*. $\mathcal{B}$ runs algorithm $TAKeyGen(PK, MSK, \Gamma_{u_2})$ to generate u_2's keys (DSK_{u_2}, KEK_{u_2}) and returns to $\mathcal{A}$.

Challenge: $\mathcal{A}$ submits two equal length messages M_0 and M_1 to $\mathcal{B}$. $\mathcal{B}$ chooses a random $\hat{b} \in \{0,1\}$, and runs the $TAEncrypt(CT, PK, MSK)$, and $DOEncrypt(OBDD, PK, M)$ algorithm to obtain $(Hdr^*, \hat{CT}^*)$. Then, $\mathcal{B}$ sends $(Hdr^*, \hat{CT}^*)$ to $\mathcal{A}$.

Phase 2: This phase is similar to Phase 1.

Guess: $\mathcal{A}$ outputs a guess $\hat{b}^* \in \{0,1\}$. $\mathcal{A}$ wins the game if $\hat{b} = \hat{b}^*$. The advantage of the $\mathcal{A}$ attack proposed scheme is $Adv_{\mathcal{A}} = |\Pr[\hat{b} = \hat{b}^*] - \frac{1}{2}|$

Definition 1. *Our proposal is CPA secure if $Adv_{\mathcal{A}}$ is negligible for any probabilistic polynomial time adversary.*

2.6 System Model

The system architecture of the scheme mainly comprises five entities, as shown in Figure 4.

TA: It maintains the KEK tree and distributes the associated keys to the users. It also reencrypts the ciphertext and updates public keys and private keys of the revoked attributes.

CS: The CS provides services for accessing and downloading ciphertext for DU and data storage services for DO.

DS: It is a server with powerful computing power, which plays a vital role in alleviating the complex computational burden on DU.

The DU sends part of the ciphertext and the conversion key to the DS. The DS generates the conversion ciphertext and then sends it to the DU. DU only needs a simple exponential to get the plaintext information. This way of outsourcing decryption largely reduces the burden of decryption calculation for users.

DO: The DO is the owner of the message, who can set an access policy for the message. The message was encrypted into ciphertext under the access policy. DO upload ciphertext to the CS for the visitor to download.

DU: The DU is a visitor of the message. The attribute private key of each data user is distributed by TA. The DU is also responsible for downloading the ciphertext from the CS, generating the conversion key, and sending the partial ciphertext to the DS.

The relevant symbols and their meanings are given, as described in Table 1. The scheme contains the following eight algorithms.

Setup$(1^\theta) \rightarrow (PK, MSK)$**:** The algorithm is executed by the TA. It inputs a security parameter θ and outputs public parameter PK and master key MSK.

TAKeyGen$(PK, MSK, \Gamma) \rightarrow (DSK, KEK_i)$**:** In the algorithm, the TA generates private keys DSK and the $KEKs$ for the user. It inputs the master key MSK, the public key PK, and the user?s attribute set Γ.

DOEncrypt$(OBDD, PK, M) \rightarrow CT$**:** The data user runs the algorithm to encrypt plaintext M. It inputs the public key PK, message M, and $OBDD$ access structure. It generates a ciphertext CT.

GenTK$(DSK, KEK_i) \rightarrow (TK, RK)$**:** This algorithm inputs the private keys DSK and KEK_i. It outputs transformation keys TK and the corresponding retrieving keys RK.

TAEncrypt$(CT, PK, MSK) \rightarrow (Hdr, \hat{CT})$**:** The TA executes the encryption algorithm. It inputs the ciphertext CT, the public key PK, and the private keys MSK. It outputs a ciphertext $\hat{CT}$ and Hdr.

DSDecrypt$(PK, TK, CT') \rightarrow TT$**:** The decryption algorithm is executed by the DS. It inputs the public parameters PK, the transformation keys TK, and the ciphertext CT'. It produces a transformation of ciphertext TT.

DUDecryptVerify$(RK, TT) \rightarrow (M, \perp)$**:** The DU runs the decryption algorithm and verifies the correctness of the ciphertext. The algorithm is operated by DU with inputs TT, RK and outputs M or $\perp$.

TAReEncrypt$(\hat{CT}, Hdr) \rightarrow (CT^*, Hdr^*)$**:** The algorithm of re-encrypted the ciphertext is performed by TA when attribute revocation occurs. Then upload ciphertext to CSP.

3 Construction of the proposed scheme

For each attribute in the system, the TA randomly chooses $t_1^+, t_2^+, t_3^+, ..., t_i^+, t_1^-, t_2^-, t_3^-, ..., t_i^- \in Z_p^*$, where t_i^+, t_i^- corresponds with the positive value and negative value of attribute i.

Setup$(1^\theta) \rightarrow (PK, MSK)$**:** The TA randomly chooses $\gamma \in Z_p^*$ and computes $Y = \hat{e}(g,g)^\gamma$. For each attribute in the system, TA randomly selects $t_1^+, t_2^+, t_3^+, ..., t_i^+, t_1^-, t_2^-, t_3^-, ..., t_i^- \in Z_p^*$, where t_i^+ and t_i^- are the positive and negative values of corresponding of i. In this paper, $\hat{t_i}$ represents the positive and negative value of the attribute i. When it is a positive value, it is denoted by $\hat{t_i} = t_i^+$. Otherwise, it is $\hat{t_i} = t_i^-$. The algorithm outputs the public key $PK = \{g, \hat{e}(g,g)^\gamma, \hat{T_i} = g^{\hat{t_i}}\}$, $MSK = \{\gamma, t_1^+, t_2^+, t_3^+, ..., t_i^+, t_1^-, t_2^-, t_3^-, ..., t_i^-\}$.

TAKeyGen$(PK, MSK, \Gamma) \rightarrow (DSK, KEK_i)$**:** Γ is the users' attribute set. TA randomly chooses $\eta \in Z_p^*$ and computes $KP_1 = g^{\eta+\gamma}$, $KP_2 = g^{\eta+\sum_{i\in\Gamma}\hat{t_i}\eta}$, $KP_3 = g^{\eta}$ for each user. The private key of the user with the attribute set Γ is $DSK = \{KP_1, KP_2, KP_3\}$. TA generates the attribute group key for the user using the KEK tree described above. The TA randomly chooses ζ_j and computes $KEK_i = \hat{T_i}^{\eta\zeta_j} = g^{\hat{t_i}\eta/\zeta_j}$, where attribute $i \in \Gamma$.

DOEncrypt$(OBDD, PK, M) \rightarrow CT$**:** The DO defined $OBDD = \{Node(id, i, high(id), low(id)) \mid id \in ID,\ i \in I\}$. We specify the number of valid paths as t, and define all valid paths as $Path = \{Path_1, Path_2, Path_3, ..., Path_t\}$. The DO selects randomly $s \in Z_p^*$ and compute $C_M = MY^s$, $C_s = g^s$, $\forall i \in O : C'_{Path_i} = \prod \hat{T_i}^{(-s)}$, a validation mark $VK = e(g,g)^{\gamma s}$. The DO outputs the ciphertext as $CT = \{C_M, C_s, C'_{Path_i}\}$ and a validation mark VK for uploading to CS.

GenTK$(DSK, KEK_i) \rightarrow (TK, RK)$**:** The algorithm is run by the DU. The DU sends TK to DS as the transformation key, and RK is saved by the user as the retrieving key. The DU selects randomly $\sigma \in Z_p^*$ and computes TK, RK, where $TK = \{\ KP_2^{1/\sigma}, KP_3^{1/\sigma}, KEK_i^{1/\sigma}\}$, $RK = \sigma$.

TAEncrypt$(CT, PK, MSK) \rightarrow (Hdr, \hat{CT})$**:** The TA re-encrypts the ciphertext sent by the user and generates a new ciphertext $\hat{CT}$ and ciphertext header Hdr. TA randomly selects $\mu_i \in Z_p^*$ and computes $C_{\hat{Path_i}} = C'_{Path_i} g^{\sum \mu_i}$, $\hat{CT} = \{C_M, C_s, C_{\hat{Path_i}}\}$, $Hdr = \{i, \zeta_j, E(\mu_i) = g^{\mu_i \zeta_j / \hat{t_i}}\}$.

DSDecrypt$(PK, TK, CT') \rightarrow TT$**:** The DU first obtains the ciphertext $\hat{CT}$ from CS, and then sends the conversion key TK and ciphertext CT' to the DS. The DS partially decrypts CT' to generate the conversion ciphertext, where CT' is (C_s, C'_{path_i}), and the calculation process of the conversion ciphertext is as follows.

$$
\begin{aligned}
T_1 &= \hat{e}(C_s, KP_1^{1/\sigma}) \\
&= \hat{e}(g,g)^{s(\eta+\gamma)/\sigma} \\
T_2 &= \frac{\hat{e}(C_s, KP_2^{1/\sigma})\hat{e}(KP_3^{1/\sigma}, C'_{Path_i})}{\prod_{i\in O} e\left(KEK_i^{1/\sigma}, Hdr\right)} \\
&= \frac{\hat{e}(g^s, g^{(\sum \hat{t_i}\eta+\eta)/\sigma})\hat{e}(g^{\eta/\sigma}, g^{-s\sum \hat{t_i}} g^{\sum \mu_i})}{\prod_{i\in O} \hat{e}(g^{\hat{t_i}\eta/\zeta_j\sigma}, g^{\mu_i\zeta_j/\hat{t_i}})} \\
&= \hat{e}(g,g)^{s\eta/\sigma}
\end{aligned}
$$

DUVerifyDecrypt$(RK, TT) \rightarrow (M, \perp)$**:** DU first verifies the correctness of the converted ciphertext TT, if $VK \neq (T_1/T_2)^{\sigma}$, it indicates that the converted ciphertext verification fails and outputs the termination symbol $\perp$. if $VK = (T_1/T_2)^{\sigma}$, it indicates that the calculation result returned by DS is correct and the user can proceed to the last step of the decryption operation, the decryption calculation formula is as follows.

$$
\begin{aligned}
M &= C_M/(T_2/T_1)^{\sigma} \\
&= M\hat{e}(g,g)^{\gamma s}/(\frac{\hat{e}(g,g)^{s\eta/\sigma}}{\hat{e}(g,g)^{s(\gamma+\eta)/\sigma}})^{\sigma}
\end{aligned}
$$

If a user's attribute att_f is revoked, keys for attribute att_f of other users must be updated. TA updates the public key and keys of attribute att_f. The ciphertext associated with attribute att_f has to be re-encrypted as well. The TA selects an exponent at random $\rho_f \in Z_p^*$ and computes $\overline{\hat{T_f^*}} = \hat{T}_f^{*\rho_f}, \overline{\hat{t_f^*}} = \hat{t_f^*}\rho_f$. The public key of the attribute is updated $\overline{\hat{T_f^*}}$ and the private key is $\overline{\hat{t_f^*}}$. Next, the TA renews the attribute group as $Mincs(\hat{AG}_i)$. If $AG_i = \{u_1, u_2, u_3\}$, then $Mincs(AG_i) = \{v_4, v_1\}$. Assuming u_3 is revoked from AG_i, then $Mincs(\hat{AG}_i) = \{v_4\}$. Immediately after, the TA performs an intersection of $Path(u_k)$ and $Mincs(\hat{AG}_i)$.

TAReEncrypt$(\hat{CT}, Hdr) \rightarrow (CT^*, Hdr^*)$**:** The TA chooses a random $s^*, \mu_i^*, \zeta_j^* \in Z_p^*$ and re-encrypts the ciphertext as

$$
\begin{aligned}
CT^* &= (C_M^*, C_s^*, C_{\hat{Path_i}}^*), C_M^* = C_M\hat{e}(g,g)^{\gamma s^*}, \\
C_s^* &= C_s g^{s^*}, C_{\hat{Path_i}}^* = g^{-(s+s^*)\sum \hat{t_i}} g^{\sum \mu_i}
\end{aligned}
$$

$$
Hdr^* = \begin{Bmatrix} \zeta_j, E(\mu_i) = g^{\mu_i\zeta_j/\hat{t_i}}, & \zeta_j \in Mincs(AG_i) \\ \zeta_j^*, E(\mu_i^*) = g^{\mu_i^*\zeta_j^*/\hat{t_f^*}}, & \zeta_j^* \in Mincs(\hat{AG}_i) \end{Bmatrix} \quad (1)
$$

3.1 Security Analysis

The challenge of our proposal is to prevent collusion attacks between existing users and revoked users. Our scheme is resistant to conspiracy attacks because every user's KEK and DSK are bounded.

Theorem 1. *If there exists an arbitrary probabilistic polynomial adversary $\mathcal{A}$ to attack our scheme by ϵ advantage, we can build a simulator $\mathcal{B}$, and $\mathcal{B}$ uses $\mathcal{A}$ to attack the CDH hypothesis. Specifically, suppose that $\mathcal{A}$ can attack our scheme with a non-negligible advantage in ϵ after making q_1 Type-A queries and q_2 Type-B queries. Then, we enable construct $\mathcal{C}$ to attack the CDH hypothesis with a non-negligible advantage $\frac{\epsilon}{q_1 q_2}$.*

Proof. During the game interaction, to maintain consistency and resist collusion attacks, the simulator $\mathcal{C}$ keeps two lists $List_{u_1}$ and $List_{u_2}$ storing the results of previous queries. $\mathcal{B}$ and $\mathcal{A}$ can simulate the game interaction as follows.

Table 3: Comparisons of communication costs

Scheme	$\|PK\|$	$\|MSK\|$	$\|SK\|$	$\|CT\|$	$\|\hat{CT}\|$
[8]	$2\|g\|+2\|g_T\|$	$\|p\|+\|g\|$	$(2N_s+1)\|g\|$	$(2N_l+1)\|g\|+\|g_T\|$	$(3N_l+1)\|g\|+\|g_T\|$
[11]	$(N_a+2)\|g\|+\|g_T\|$	$(N_a+1)\|p\|+\|g\|$	$(3N_s+1)\|g\|$	$(2N_l+1)\|g\|+\|g_T\|$	$(3N_l+1)\|g\|+\|g_T\|$
[12]	$(2N_a+1)\|g\|+\|g_T\|$	$(2N_a+1)\|p\|$	$2\|g\|$	$(N_r+1)\|g\|+\|g_T\|$	/
[5]	$(4N_a+1)\|g\|+\|g_T\|$	$(2N_a+1)\|p\|$	$(3N_k+1)\|g\|$	$(2N_r+1)\|g\|+\|g_T\|$	$(2N_r+N_l+1)\|g\|+\|g_T\|$
Ours	$(2N_a+1)\|g\|+\|g_T\|$	$(2N_a+1)\|p\|$	$(N_k+3)\|g\|$	$(N_r+1)\|g\|+\|g_T\|$	$(N_r+N_l+1)\|g\|+\|g_T\|$

Init: The Challenger $\mathcal{C}$ sends $A=g^{z_1}, B=g^{z_2}, z_1, z_2 \in Z_p^*$ to the $\mathcal{B}$. The challenge access policy O^* and the challenge attribute att_f^* are selected by $\mathcal{A}$ and sent to the simulator $\mathcal{B}$.

Setup: $\mathcal{B}$ picks random number $\gamma, t_i^+, t_i^- \in Z_p^*$ and computes $\hat{e}(g,g)^{\gamma}, \hat{T}_i = g^{\hat{t_i}}$. For challenge attribute att_f^*, $\mathcal{B}$ randomly picks $t_f^*, t_f^{*'}$ and calculates $\hat{T_f^*} = g^{\hat{t_f^*}}$. When the attribute att_f^* is revoked, $\mathcal{B}$ updates the public key and secret key to att_f^*. $\mathcal{B}$ computes $\overline{\hat{t_f^*}} = \hat{t_f^*} z_1$, and $\overline{\hat{T_f^*}} = g^{\hat{t_f^*} z_1} = A^{\hat{t_f^*}}$.

Phase 1: $\mathcal{B}$ maintains two lists named a $List_{u_1}$ and $List_{u_2}$. Initially, both are empty. The following queries are issued by $\mathcal{A}$.

1) Type-A, u_1 satisfies access policy O^*, but the attribute att_f^* has been revoked. $\mathcal{B}$ first checks whether u_1 exists in $List_{u_1}$. If it exists, $\mathcal{B}$ sends $(u_1, r_1, DSK_{u_1}, KEK_{u_1})$ to $\mathcal{A}$, otherwise $\mathcal{B}$ computes $(u_1, r_1, DSK_{u_1}, KEK_{u_1})$. $\mathcal{B}$ firstly chooses a random exponent $\eta_1 \in Z_p^*$.

$$\begin{aligned} DSK_{u_1} &= \{KP_1 = g^{z_2(\sum t_i \eta_1 + \eta_1)}, \\ &\quad KP_2 = g^{z_2\eta_1}, KP_3 = g^{z_2\eta_1+\gamma}\} \\ KEk_{u_1} &= \{\forall att_i \in \{\Gamma_{u_1} - att_f^*\} : \zeta_j, \\ &\quad KEK_i = g^{\hat{t_i}\eta_1/\zeta_j} att_i = att_f^* : \zeta_j^*, \\ &\quad KEK_f^* = g^{z_2\eta_1 \hat{t_f^*}/\zeta_j} = B^{\eta_1 \hat{t_f^*}/\zeta_j^*}\} \end{aligned}$$

2) Type-B, u_2 does not satisfy access policy O^*, but has the attribute att_f^*. $\mathcal{B}$ first checks whether u_2 exists in $List_{u_2}$. If it exists, $\mathcal{B}$ sends $(u_2, r_2, DSK_{u_2}, KEK_{u_2})$ to $\mathcal{A}$, Otherwise $\mathcal{B}$ computes $(u_2, r_2, DSK_{u_2}, KEK_{u_2})$. $\mathcal{B}$ firstly chooses a random exponent $\eta_2 \in Z_p^*$.

$$\begin{aligned} DSK_{u_2} &= \{KP_1 = g^{\sum t_i \eta_2 + \eta_2}, KP_2 = g^{\eta_2}, \\ &\quad KP_3 = g^{\eta_2+\gamma}\} \\ KEk_{u_2} &= \{\forall att_i \in \{\Gamma_{u_2} - att_f^*\} : \zeta_j, \\ &\quad KEK_i = g^{\hat{t_i}\eta_2/\zeta_j} att_i = att_f^* : \zeta_j^*, \\ &\quad KEK_f^* = A^{\hat{t_f^*}\eta_2/\zeta_j^*}\} \end{aligned}$$

Challenge: After completing the first phase, adversary $\mathcal{A}$ submits two messages of the same size messages M_0 and M_1. $\mathcal{B}$ chooses a random $s, \hat{b} \in \{0,1\}$ and calculates the ciphertext.

$$\begin{aligned} CT^* &= \{C_M^* = M_{\hat{b}} e(g,g)^{\gamma s}, C_s^* = g^s, \\ &\quad \hat{C_{Path_i}^*} = g^{-(\hat{t_1}+\hat{t_2}+\ldots\hat{t_i}+z_1 t_f^*)s} \cdot \\ &\quad g^{(\mu_1+\mu_2+\ldots+z_1\mu_f^*)}\} \\ Hdr^* &= \{\forall i \in \{\Gamma - att_f^*\}, E(\mu_i) = g^{\mu_i \zeta_j/\hat{t_i}} \\ &\quad att_i = att_f^*, E^*(\mu_i) = g^{z_1\mu_f^*\zeta_j^*/\hat{t_f^*}}\} \end{aligned}$$

$\mathcal{B}$ sends (Hdr^*, CT^*) to the adversary $\mathcal{A}$.

Phase 2: Similar to Phase 1, adversary $\mathcal{A}$ continues to ask $\mathcal{B}$ for the private key. The restriction is that the set of interrogated attribute private keys does not meet the access policy.

Guess: The adversary $\mathcal{A}$ outputs a value $\hat{b}^*$ as a guess for $\hat{b}$. Suppose the adversary $\mathcal{A}$ guesses $\hat{b} = \hat{b}^*$, and the $\mathcal{B}$ chooses an $(u_1, r_1, DSK_{u_1}, KEK_{u_1})$ and an $(u_2, r_2, DSK_{u_2}, KEK_{u_2})$ from $List_{u_1}$ and $List_{u_2}$, respectively. For att_f^*, there exists DSK_{u_1} in $List_{u_1}$ and KEK_{u_2} in $List_{u_1}$ combined such that the following equation holds.

$$\begin{aligned} &\frac{\hat{e}(C_s^*, KP_2)\hat{e}(\hat{C_{Path_i}^*}, KP_3)}{\prod_{i\in S_2}\hat{e}(KEK_i^*, E^*(\mu_i))} \\ &= \frac{e(g,g)^{s\eta_1} e(g^{\mu_1+\mu_2+\ldots+z_1\mu_f^*\mu_f^*}, B^{\eta_1})}{e(g^{\eta_2}, g^{\mu_1+\mu+2+\ldots+z_1\mu_f^*})} \\ &= e(g,g)^{s\eta_1} \end{aligned}$$

Only if $g^{z_1 z_2} = g^{\frac{\eta_1\eta_2}{z_1}}$, $\mathcal{B}$ can get $g^{z_1 z_2}$ and output $(KEK_f^*)^{\frac{\eta_1\zeta_j^*}{t_f^* z_1^2}}$ as its result.

Suppose $\mathcal{A}$ executes q_1 Type-A queries and q_2 Type-B queries. The probability that $\mathcal{B}$ correctly selects $(u_1, r_1, KEK_{u_1}, DSK_{u_1})$ and $(u_2, r_2, KEK_{u_2}, DSK_{u_2})$ from $List_{u_1}$ and $List_{u_2}$. Therefore, the advantage of $\mathcal{A}$ in breaking the CDH hypothesis is $\frac{\epsilon}{q_1 q_2}$.

□

4 Performance Evaluation

In this part, we analyze the functionality, communication overhead, and computation overhead of our scheme. We also compare it with existing literature.

Table 4: Comparisons of computation costs

Scheme	Setup	KeyGen	Encryption	Reencryption	Decryption
[8]	$T_e + T_p$	$(2N_s + 2)T_p$ $+(N_s + 1)T_m$	$(2N_d + 2)T_p + T_m + N_d T_H$	$N_l T_p$	$(2N_k + 1)T_e$ $+(2N_k + 2)T_m$
[11]	$T_e + 3T_p + T_m$	$(2N_s + 2)T_p$ $+(N_s + 1)T_m + N_s T_H$	$(2N_d + 2)T_p + T_m + N_d T_H$	$2N_l T_p + N_l T_m$	$(3N_k + 1)T_e$ $+(2N_k + 2)T_m$
[12]	$T_e + (N_a + 1)T_p$	$2T_p + T_m$	$(N_r + 2)T_p + (N_r N_a + 1)T_m$	/	$2T_e + 2T_m$
[5]	$T_e + (2N_a + 1)T_p$	$4N_k T_p + 4N_k T_m$	$(2N_r N_a + 2)T_p + (2N_r N_a + 1)T_m$	$2N_l T_p + 3N_l T_m$	$(3N_k + 1)T_e$ $+(2N_k + 2)T_m$
Ours	$T_e + (N_a + 1)T_p$	$(2N_k + 2)T_p + N_k T_m$	$(N_r + 2)T_p + (N_r N_a + 1)T_m$	$2N_l T_p + N_l T_m$	$T_p + T_m$

1) Functionality:
The results of the functional analysis of our scheme and related schemes are shown in Table 3. We discovered that there are very few studies on CP-ABE based on OBDD in cloud environments after our extensive review. We only found two pieces of literature, [12] and [5]. In addition, we exemplify [8] and [11] that are relevant to our work. They are both CP-ABE schemes constructed based on access trees. [5, 8, 11] and our scheme solved the attribute revocation. [5, 11] and our scheme achieve the problem of resisting user complicity. [8] and [12] do not resist user complicity. In terms of implementing outsourced decryption and verifiability, only our solutions support outsourcing and verifiability. [12] is the first scheme that proposes to use OBDD in CP-ABE. But it is a basic CP-ABE scheme, which does not have an attribute revocation, resistance to user complicity, and outsourcing computing functions.

2) Communication Overhead:
We compared the communication cost of our scheme with the literature [5, 8, 11, 12]. This result is shown in Table 4. The communication cost mainly comes from public key PK and private key MSK of the TA, private key of the user SK, and ciphertext CT and re-encrypted ciphertext $\hat{CT}$ of the cloud service. For the convenience of the discussion, let $|p|$, $|g|$, and $|g_T|$ be the element sizes of Z_p, g, and g_T, respectively. N_a denotes the number of attributes in the system, and N_k indicates the number of users' attribute keys. N_l denotes the number of attributes in the ciphertext and N_r denotes the number of valid paths. N_s represents the number of users' attributes. N_d represents the number of attributes that satisfy the access policy. From the analysis in Table 4, it is concluded that the literature [5] has the largest communication overhead in PK. [12] is consistent with our scheme regarding the communication overhead of PK, followed by [11]. The cost of [8] is the lowest. This is since [8] and [11] are schemes constructed based on access trees, which are less flexible on access policy expressiveness. Although the communication overhead of [12] and [5] is the largest compared to other schemes in PK, the size of this overhead is reasonably acceptable. In SK, the cost of [12] is the lowest because it is a most basic scheme that does not implement attribute revocation. It also is the lowest in CT. The communication overhead of [8] and [11] in SK is linearly related to the number of attributes owned by the user, [5] and our scheme is related to the number of private keys distributed to the user. The overheads of [8] and [11] in CT are linearly related to the number of leaf nodes of the access tree. The overheads of [5] and our scheme in CT are linearly related to the valid paths. Since [12] does not support attribute revocation, there is also no communication overhead for the re-encrypted ciphertext $\hat{CT}$. Our scheme has the lowest communication overhead in re-encryption. Obviously, the overhead of our scheme in SK, CT and $\hat{CT}$ is on the small side in CP-ABE constructed based on OBDD.

3) Computation Overhead:
The computational overhead operations mainly include exponentiation, bilinear mapping, multiplication and hash. T_e denotes the time to perform a bilinear mapping operation. T_p represents the time to execute an exponential operation. T_m means the time to complete a multiplication operation. T_H denotes the time to perform a hash operation. As we can see from Table 5, [8] has the lowest computing overhead during the Setup, followed by [11]. Our solution is consistent with the computational overhead of [12]. In KeyGen, the computation overhead of [8] and [11] grows with the growth of user-owned attributes. The computational overhead of [5] and our scheme grows with the valid path. In Encryption, the computation overhead of [8] and [11] correlates with the number of leaf nodes of the access tree. The computational overhead of [12] and [5] and our scheme is related to the number of valid paths. In Re-encryption, [12] does not support attribute revocation, so there is no computational overhead from re-encryption. The computation overhead of [8] is the lowest, followed by [11] and our scheme. In Decryption, the computation overhead of our scheme is much less than the other schemes.

The Pairing-Based Cryptography (PBC) library's type A curves are used in the experimental code, which is written in Golang. One multiplication, one bilinear mapping operation, and one exponentiation operation are measured. Table 5 presents the result. This paper mainly simulates the computational overheads of Setup, TAKeyGen, DOEncrypt, GenTK, TAEncrypt, DSDecrypt and DUDecryptVerify. The number of attributes is selected

Table 5: The computational overhead of each operation

Symbols for operations	T_e	T_p	T_m
The time of operation (ms)	1.296	1.078	0.004

as 3, 6, 9, 12, 15, and run 200 times. Then, the average value is taken as the final value. As seen in the graphic, the time overhead of TAKeyGen, GenTK, TAEncrypt and DSDecrypt is linearly related to the number of attributes. In our simulations, the number of valid paths is constant, so the computational overhead of DOEncrypt is not impacted by the number of attributes. DSDecrypt is processed by a decryption server with high computing power. We can disregard it. The computational time of the DUDecryptVerify is very low, which is due to the fact that the complex computations are executed by the decryption service. The DO only needs a small computation and verification. The computation burden of the user is significantly reduced to about 1ms, as seen in Figure 5(g).

5 Conclusion

To our knowledge, we first propose a CP-ABE revocable outsourcing decryption scheme based on the OBDD. OBDD is more flexible and expressive compared to traditional access structures. Our proposal can effectively revoke attributes and resist user complicity. It was proven to be CPA secure under the CDH difficulty problem. Besides, we outsource the complex computations to the DS to reduce the user's burden. Moreover, the user doesn't need extra keys to verify the transformation of ciphertext. According to a theoretical analysis, our scheme has a lower computational and communication cost. In future work, we will construct a scheme for multiple authorities with OBDD in CP-ABE.

Acknowledgments

This work was supported by the National Natural Science Foundation of China under Grant 62072369, 62072371, 62072078, 61802303 and 61772418, the Key Research and Development Program of Shaanxi under Grant 2020ZDLGY08-04, the Innovation Capability Support Program in Shaanxi Province of China under Grant 2020KJXX-052, the Basic Research Program of Qinghai Province under Grant 2020-ZJ-701, Yinghui Zhang is supported by the Shaanxi Special Support Program Youth Top-notch Talent Program, Qianqian Zhao is supported by the Natural Science Foundation of Shanghai under Grant 19ZR1454100.

References

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

[2] N. Chen, J. Li, Y. Zhang, and Y. Guo, "Efficient cp-abe scheme with shared decryption in cloud storage," *IEEE Transactions on Computers*, vol. 71, no. 1, pp. 175–184, 2022.

[3] Y. Chen, L. Song, and G. Yang, "Attribute-based access control for multi-authority systems with constant size ciphertext in cloud computing," *China Communications Journal*, vol. 13, no. 2, 2016.

[4] H. Cui, R. H. Deng, J. Lai, X. Yi, and S. Nepal, "An efficient and expressive ciphertext-policy attribute-based encryption scheme with partially hidden access structures, revisited," *Computer Networks*, vol. 133, pp. 157–165, 2018.

[5] K. Edemacu, B. Jang, and J. W. Kim, "Collaborative ehealth privacy and security: An access control with attribute revocation based on obdd access structure," *IEEE journal of biomedical and health informatics*, vol. 24, no. 10, pp. 2960–2972, 2020.

[6] K. Gu, W. B. Zhang, X. Li, and W. J. Jia, "Self-verifiable attribute-based keyword search scheme for distributed data storage in fog computing with fast decryption," *IEEE Transactions on Network and Service Management*, 2021.

[7] Y. Hou, S. Garg, L. Hui, D. N. K. Jayakody, R. Jin, and M. S. Hossain, "A data security enhanced access control mechanism in mobile edge computing," *IEEE Access*, vol. 8, pp. 136 119–136 130, 2020.

[8] J. Hur and D. K. Noh, "Attribute-based access control with efficient revocation in data outsourcing systems," *IEEE Transactions on Parallel and Distributed Systems*, vol. 22, no. 7, pp. 1214–1221, 2010.

[9] M. S. Hwang, C. C. Lee, and S. T. Hsu, "An elgamal-like secure channel free public key encryption with keyword search scheme," *International Journal of Foundations of Computer Science*, vol. 30, no. 02, pp. 255–273, 2019.

[10] J. Li, Y. Wang, Y. Zhang, and J. Han, "Full verifiability for outsourced decryption in attribute based encryption," *IEEE transactions on services computing*, vol. 13, no. 3, pp. 478–487, 2017.

[11] J. Li, W. Yao, J. Han, Y. Zhang, and J. Shen, "User collusion avoidance cp-abe with efficient attribute revocation for cloud storage," *IEEE Systems Journal*, vol. 12, no. 2, pp. 1767–1777, 2017.

[12] L. Li, T. Gu, L. Chang, Z. Xu, Y. Liu, and J. Qian, "A ciphertext-policy attribute-based encryption based on an ordered binary decision diagram," *IEEE Access*, vol. 5, pp. 1137–1145, 2017.

[13] Z. Li, W. Li, Z. Jin, H. Zhang, and Q. Wen, "An efficient abe scheme with verifiable outsourced encryption and decryption," *IEEE Access*, vol. 7, pp. 29 023–29 037, 2019.

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

Table 6: OBDD access structural expressions

Access Structure	Operations supported				Positive and negative values
	AND	OR	NOT	Thresold	
AND gates	×	√	√	√	×
Access tree	×	×	√	×	√
LSSS	×	×	√	×	√
OBDD	×	×	×	×	×

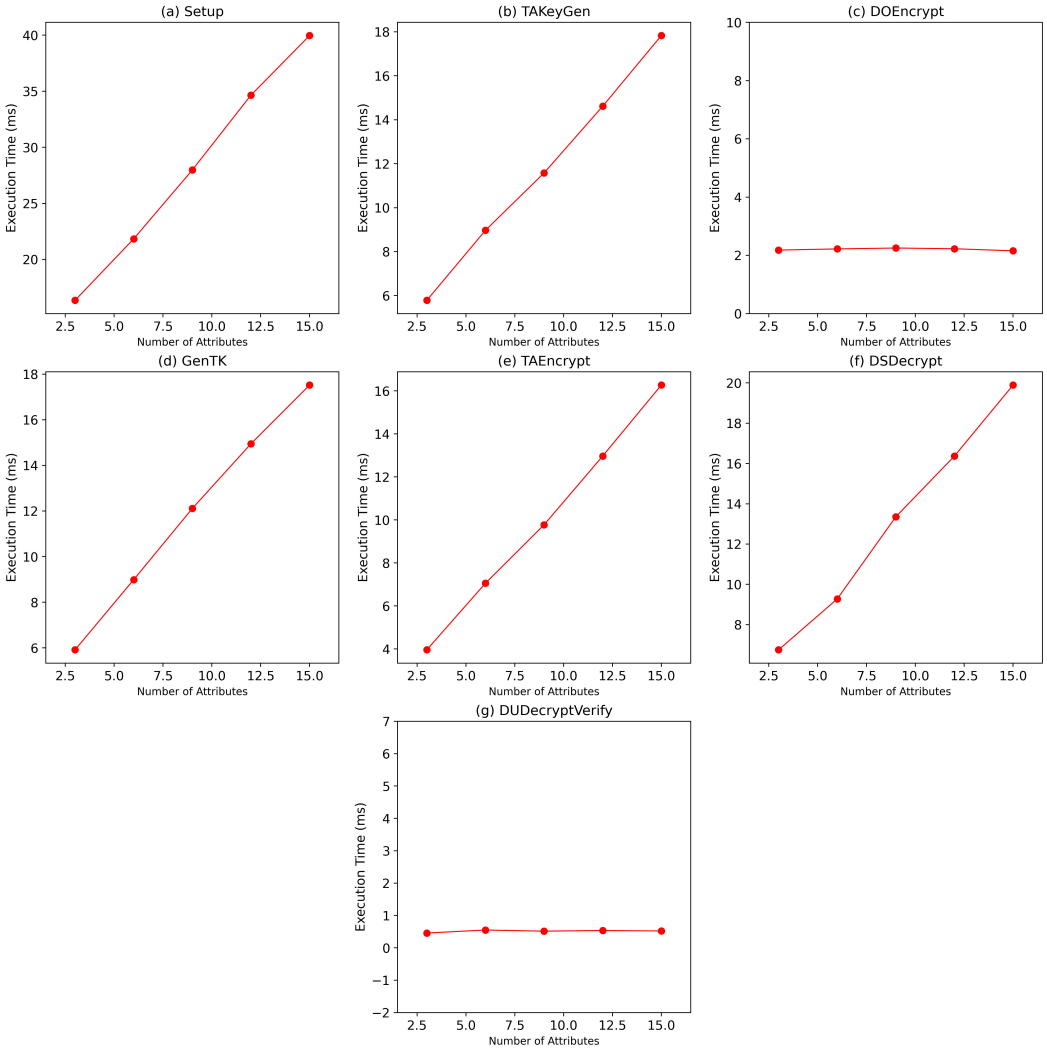


Figure 5: Experimental results of the computational and communication costs

[15] J. K. Liu, T. H. Yuen, P. Zhang, and K. Liang, "Time-based direct revocable ciphertext-policy attribute-based encryption with short revocation list," in *International Conference on Applied Cryptography and Network Security*. Springer, pp. 516–534, 2018.

[16] L. Liu, Z. Cao, "Analysis of a privacy preserving ranked multi-keyword search scheme," *International Journal of Electronics and Information Engineering*, vol. 12, no. 2, pp. 53–59, 2020.

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

[18] M. Lyu, X. Li, and H. Li, "Efficient, verifiable and privacy preserving decentralized attribute-based encryption for mobile cloud computing," in *2017 IEEE Second International Conference on Data Science in Cyberspace (DSC'17)*, pp. 195–204, 2017.

[19] F. Meng, L. Cheng, and M. Wang, "Ciphertext-policy attribute-based encryption with hidden sensitive policy from keyword search techniques in smart city," *EURASIP Journal on Wireless Communications and Networking*, vol. 2021, no. 1, pp. 1–22, 2021.

[20] T. V. X. Phuong, G. Yang, and W. Susilo, "Hidden ciphertext policy attribute-based encryption under standard assumptions," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 1, pp. 35–45, 2016.

[21] B. Qin, Q. Zhao, D. Zheng, and H. Cui, "(dual) server-aided revocable attribute-based encryption with decryption key exposure resistance," *Information Sciences*, vol. 490, pp. 74–92, 2019.

[22] W. Susilo, G. Yang, F. Guo, and Q. Huang, "Constant-size ciphertexts in threshold attribute-based encryption without dummy attributes," *Information Sciences*, vol. 429, pp. 349–360, 2018.

[23] W. Wang, G. Zhang, and Y. Shen, "A cp-abe scheme supporting attribute revocation and policy hiding in outsourced environment," in *2018 IEEE 9th International Conference on Software Engineering and Service Science (ICSESS'18)*, pp. 96–99, 2018.

[24] J. Zhang, Z. Cheng, X. Cheng, and B. Chen, "Oachas: outsourced access control with hidden access structures in fog-enhanced iot systems," *Connection Science*, vol. 33, no. 4, pp. 1060–1076, 2021.

[25] Y. Zhang, X. Chen, J. Li, H. Li, and F. Li, "Attribute-based data sharing with flexible and direct revocation in cloud computing," *KSII Transactions on Internet and Information Systems (TIIS)*, vol. 8, no. 11, pp. 4028–4049, 2014.

[26] Y. Zhang, D. Zheng, R. Guo, and Q. Zhao, "Fine-grained access control systems suitable for resource-constrained users in cloud computing," *Computing and Informatics*, vol. 37, no. 2, pp. 327–348, 2018.

[27] Y. Zhang, T. Zhu, R. Guo, S. Xu, H. Cui, and J. Cao, "Multi-keyword searchable and verifiable attribute-based encryption over cloud data," *IEEE Transactions on Cloud Computing*, 2021.

[28] Z. Zhang, C. Li, B. B. Gupta, and D. Niu, "Efficient compressed ciphertext length scheme using multi-authority cp-abe for hierarchical attributes," *IEEE Access*, vol. 6, pp. 38 273–38 284, 2018.

[29] J. Zhao and H. Gao, "Lsss matrix-based attribute-based encryption on lattices," in *2017 13th International Conference on Computational Intelligence and Security (CIS'17)*, pp. 253–257, 2017.

Biography

Li Chen is with the School of Cyberspace Security, National Engineering Laboratory for Wireless Security, Xi'an University of Posts and Telecommunications, Xi'an 710121, China.

Rui Guo is with the School of Cyberspace Security, National Engineering Laboratory for Wireless Security, Xi'an University of Posts and Telecommunications, Xi'an 710121, China.

Lei Xu is with the School of Cyberspace Security, National Engineering Laboratory for Wireless Security, Xi'an University of Posts and Telecommunications, Xi'an 710121, China.

Xin Wei is with the School of Cyberspace Security, National Engineering Laboratory for Wireless Security, Xi'an University of Posts and Telecommunications, Xi'an 710121, China.

Geng Yang is with the School of Cyberspace Security, National Engineering Laboratory for Wireless Security, Xi'an University of Posts and Telecommunications, Xi'an 710121, China.

Chaoyuan Zhuang is with the School of Cyberspace Security, National Engineering Laboratory for Wireless Security, Xi'an University of Posts and Telecommunications, Xi'an 710121, China.

Qianqian Zhao is with the Shanghai Research and Development Center for Micro-Nano Electronics, Shanghai 201210, China.

Multi-bit Functional Encryption for Inner Product Predicate over Lattice

Mingming Jiang, Qihong Chen, Yuyan Guo, and Dongbing Zhang
(Corresponding author: Qihong Chen)

Department of Computer Science and Technology, Huaibei Normal University
Huaibei, Anhui 235000, China
Email: chenqihong1218@163.com

(Received May 27, 2022; Revised and Accepted Oct. 9, 2022; First Online Oct. 15, 2022)

Abstract

In cloud computing, lattice-based function encryption (FE) enables fine-grained access control and resists quantum attacks. However, most proposed predicate encryption (PE) schemes are based on number theoretic problems and cannot resist quantum attacks. In addition, most existing schemes over lattice are the single bit that the efficiency is low. Therefore, an efficient multi-bit inner product predicate (IPE) encryption scheme is proposed. In this paper, a plaintext matrix can be encrypted, greatly extending the plaintext space. Moreover, our scheme's security is based on learning with errors (LWE) problems and can resist quantum attacks.

Keywords: Function Encryption; Lattice; LWE; Predicate Encryption

1 Introduction

In FE, three parties are involved, the key generation center, the encryptor and the receiver. The receiver first transmits the relevant parameters(the private key and public key) to the key generation center. The key generation center transmits the private key to the receiver and the public key to the encryptor. The encryptor encrypts the plaintext with the public key and transmits the ciphertext to the receiver. And then receiver decrypts the ciphertext with the private key to obtain the plaintext information. The above is the whole process of the function encryption system. As the traditional public key encryption system is crude, the concept of function encryption has emerged [7]. Function encryption enables fine-grained control of encrypted data. Predicate encryption and attribute-based encryption (ABE) are important examples of function encryption.

In the PE scheme [14], the predicate f is used to generate secret key sk_f, and then attribute set I is used to generate the ciphertext CT. Therefore, if and only if $f(I) = 1$, the sk_f can decrypt the CT. The first inner product predicate (IPE) encryption scheme is proposed by Katz *et al.* [14]. In [14], the attribute is represented by vector $\boldsymbol{w}$, and the predicate is represented by vector $\boldsymbol{v}$. If and only if $< \boldsymbol{v}, \boldsymbol{w} >= 0$, the CT are decrypted successfully with the secret key. Compared with ABE, the PE system has the characteristic of attribute hiding. Due to the feature of attribute hiding, it also makes the construction of PE more complicated. The IPE encryption system can support CNF and DNF formulas.

ABE originated from identity-based encryption (IBE), which was first evolved from the concept of fuzzy identity-based encryption (FIBE) [21]. On this basis, two variants of attribute-based encryption are proposed, and they are key-policy attribute-based encryption (KP-ABE) [11] and ciphertext-policy attribute-based encryption (CP-ABE) [6] respectively. As a key supporting technology for data sharing in the IoT, the FE system can realize flexible application of data based on user attributes. In CP-ABE, the access policy is used to generate the ciphertext, and the user's attributes are used to generate the secret key. In contrast, in KP-ABE, the access policy is used to generate the secret key, and the user's attributes are used to generate the ciphertext. Subsequently, many predicate encryption schemes [8, 18, 22, 24] and attribute-based schemes [5, 15, 16, 20, 26] have been proposed, but these are based on bilinear groups and cannot resist quantum attacks. The rapid development of quantum computers also makes it urgent to design functional encryption scheme that can resist quantum attacks. In recent years, because the lattice cryptosystem is simple to calculate and can resist quantum attacks, it has become an important research direction for scholars. Regev [19] first proposed the learning with errors (LWE) problem. And then it is proved as hard as the shortest vector problem (SVP) in the worst case over lattice. So, there is no probability polynomial-time (PPT) algorithm that can overcome the LWE hard problem over lattice. Agrawal *et al.* [3] first proposed a lattice-based FE scheme and it can realize attribute hiding. But the FE scheme can only realize single-bit encryption.After that, more and more lattice-based functional encryption schemes have

been proposed [1, 4, 13, 23, 25]. Therefore, this paper is mainly to construct an efficient inner product predicate encryption over lattice, which can encrypt a plaintext matrix. Our scheme is proved to be chosen plaintext attack (CPA) secure in the standard model.

1.1 Our Construction

Our scheme is based on the lattice IPE of the BGN system. The scheme is multi-bit matrix encryption. First, we use the predicate vector $\boldsymbol{v}$ to generate the private key $\boldsymbol{E}$, and then we embed the public key A and an attribute vector w into the ciphertext. In the decryption process, when $<\boldsymbol{v},\boldsymbol{w}>=0$, the ciphertext can be decrypted.

1.2 Organization

In Section 2, we exemplified some symbols that will be used in this paper, and introduced the basic lattice definitions, as well as some algorithms and hard assumption over lattices. In Section 3, the definition and security model of the functional encryption scheme are introduced. In Section 4, we mainly describe the specific scheme and security proof of the scheme in detail, and then compare our scheme with other schemes in Section 4.4. In order to expand the plaintext space, in Section 5, we construct an expansion scheme. Finally, in Section 6, a summary of this paper is made.

2 Preliminaries

2.1 Notations

The ring of the integers and the real numbers are denoted by $\mathbb{Z}$ and. let denote the ring of integers modulo q and $\mathbb{Z}_q^{n\times m}$represents the set of $n\times m$ matrices with entries in $\mathbb{Z}_q$, where q is an integer and $q\geq 2$. For a matrix $\boldsymbol{A}$, $\|\boldsymbol{A}\|$and $\boldsymbol{A}^{\mathrm{T}}$ represent the Euclidean norm and the transposed matrix of $\boldsymbol{A}$ respectively. $\tilde{\boldsymbol{A}}$ represents the Gram-Schmidt orthogonalization of $\boldsymbol{A}$ and $\left\|\tilde{\boldsymbol{A}}\right\|$ represents the Gram-Schmidt norm of $\tilde{\boldsymbol{A}}$. For parameter n, the function $negl(n)$ denote negligible function of n.

2.2 Lattice

Definition 1. *Let $\boldsymbol{b}_1,\boldsymbol{b}_2\cdots\boldsymbol{b}_m\in\mathbb{R}^m$ be a set of linearly independent vectors, and a m-dimensional full rank lattice is defined as [17]:*

$$\begin{aligned}\Lambda &= L\left(\boldsymbol{b}_1,\boldsymbol{b}_2\cdots\boldsymbol{b}_m\right)\\ &= \{\boldsymbol{y}\in\mathbb{R}^m, s.t.\exists\boldsymbol{s}\in\mathbb{Z}^m, \boldsymbol{y}=\sum_{i=1}^{m}\boldsymbol{s}_i\boldsymbol{b}_i\}\end{aligned}$$

Definition 2. *Let q be a prime, a matrix $\boldsymbol{A}\in\mathbb{Z}_q^{n\times m}$ and a vector $\boldsymbol{u}\in\mathbb{Z}_q^n$, define three m-dimensional full rank lattices [19]:*

$$\Lambda_q\left(\boldsymbol{A}\right)=\left\{\boldsymbol{e}\in\mathbb{Z}^m, s.t.\exists\boldsymbol{s}\in\mathbb{Z}_q^n, \boldsymbol{A}^{\mathrm{T}}\boldsymbol{s}=\boldsymbol{e}\bmod q\right\}$$

$$\Lambda_q^{\perp}\left(\boldsymbol{A}\right)=\{\boldsymbol{e}\in\mathbb{Z}^m, s.t.\boldsymbol{A}\boldsymbol{e}=0\bmod q\}$$

$$\Lambda_q^{u}(\boldsymbol{A})=\{\boldsymbol{e}\in\mathbb{Z}^m, s.t.\boldsymbol{A}\boldsymbol{e}=\boldsymbol{u}\bmod q\}$$

If $t\in\Lambda_q^{\boldsymbol{u}}\left(\boldsymbol{A}\right)$, then $\Lambda_q^{\boldsymbol{u}}\left(\boldsymbol{A}\right)=\Lambda_q^{\perp}\left(\boldsymbol{A}\right)+t$.

2.3 Relevant Algorithms

Lemma 1. *[10]. Let a prime $q\geq 3$ and a positive integer $m=\lfloor 6n\log q\rfloor$. There is a PPT algorithm $\boldsymbol{TrapGen}(q,n)$ that output $\boldsymbol{A}\in\mathbb{Z}_q^{n\times m}$ and $\boldsymbol{T_A}\in\mathbb{Z}^{m\times m}$, where the matrix $\boldsymbol{A}$ is statistically close to uniform distribution and $\boldsymbol{T_A}$ is a basis for $\Lambda_q^{\perp}\left(\boldsymbol{A}\right)$. And $\left\|\tilde{\boldsymbol{T}}_{\boldsymbol{A}}\right\|\leq o\left(\sqrt{n\log q}\right)$, $\|\boldsymbol{T_A}\|\leq o\left(n\log q\right)$.*

Definition 3. *[12]. For any vector $\boldsymbol{c}\in\mathbb{R}^n$ and a positive parameter $s>0$, define the Gaussian distribution function on $\mathbb{R}^n$:*

$$\forall\boldsymbol{x}\in\mathbb{R}^n, \rho(\boldsymbol{x})=e^{-\pi\|(\boldsymbol{x}-\boldsymbol{c})/s\|^2}$$

Discrete Gaussian distribution function is defined over m-dimensional lattice Λ:

$$D_{\Lambda,\boldsymbol{s},\boldsymbol{c}}\left(\boldsymbol{x}\right)=\frac{\rho_{\boldsymbol{s},\boldsymbol{c}}(x)}{\rho_{\boldsymbol{s},\boldsymbol{c}}(\Lambda)}$$

Lemma 2. *[10]. For any n-dimensional lattice Λ, $\boldsymbol{c}\in span(\Lambda)$, a matrix $\boldsymbol{A}\in\mathbb{Z}_q^{n\times m}$, and $\boldsymbol{T_A}$ is a basis for $\Lambda_q^{\perp}\left(\boldsymbol{A}\right)$, $s\geq\left\|\tilde{\boldsymbol{T}}_{\boldsymbol{A}}\right\|\cdot\omega(\sqrt{\log m})$, we have*

$$\Pr\left[\boldsymbol{x}\leftarrow D_{\Lambda,s,c}:\|\boldsymbol{x}-\boldsymbol{c}\|>s\sqrt{m}\right]\leq negl(n)$$

Lemma 3. *[10]. Let $q\geq 2$ and a matrix $\boldsymbol{A}\in\mathbb{Z}_q^{n\times m}$, $\boldsymbol{T_A}$ is a basis for $\Lambda_q^{\perp}\left(\boldsymbol{A}\right)$. For $s\geq\left\|\tilde{T}_A\right\|\cdot\omega(\sqrt{\log m})$, $\boldsymbol{c}\in\mathbb{Z}^m$, $\boldsymbol{u}\in\mathbb{Z}_q^m$, here are the following algorithms:*

1) *For algorithm $Sample\Pr e(\boldsymbol{A},\boldsymbol{T_A},\boldsymbol{u},s)$, the $\boldsymbol{x}\in\Lambda_q^{u}\left(\boldsymbol{A}\right)$ sampled by the algorithm is indistinguishable from the $D_{\Lambda_q^{\boldsymbol{u}},s,\boldsymbol{c}}$.*
2) *For algorithm $SampleGaussian(\boldsymbol{A},\boldsymbol{T_A},\boldsymbol{u},s)$, the $\boldsymbol{x}\in\Lambda_q^{\perp}\left(\boldsymbol{A}\right)$ sampled by the algorithm is indistinguishable from the $D_{\Lambda,s,\boldsymbol{c}}$.*

2.4 The LWE Hard Problem

The following is the definition of noise distribution $\bar{\Psi}_{\alpha}$ [19].

Definition 4. *Let $\bar{\Psi}_{\alpha}$ means the probability distribution with mean 0 and standard deviation $\alpha/\sqrt{2\pi}$ over $\mathbb{Z}_q$, where $\alpha\in(0,1)$ and an integer $q\geq 2$. The $x\in\mathbb{R}$ be chosen from $\bar{\Psi}_{\alpha}$ and output $\lceil qx\rceil$.*

The security of our schemes is reduced to LWE problem [9].

Definition 5. *Let a prime $q\geq 2$, an integer $n\geq 1$, and a distribute $\bar{\Psi}_{\alpha}$ over $\mathbb{Z}_q$. An $(\mathbb{Z}_q,n,\bar{\Psi}_{\alpha})$ -LWE problem instance includes access to an uncertain challenge oracle*

O. *There are two possibilities for this undetermined oracle. The first is that the* O *is a noisy pseudo-random sampler* O_s*, and the second is that the oracle is a truly random sampler* $O_\$$*. Then, their specific definitions are as follows:*

1) O_s*: outputs some samples* $(\boldsymbol{A}, \boldsymbol{C}) = (\boldsymbol{A}, \boldsymbol{A}^{\mathrm{T}}\boldsymbol{S}+\boldsymbol{X}) \in \mathbb{Z}_q^{n\times m} \times \mathbb{Z}_q^{m\times m}$*, where the distribution of* $\boldsymbol{A} \leftarrow \mathbb{Z}_q^{n\times m}$ *and* $\boldsymbol{S} \leftarrow \mathbb{Z}_q^{n\times m}$ *are chosen from a uniform distribution,* $\boldsymbol{X}$ *is a sample from* $\bar{\Psi}_\alpha^{m\times m}$*.*

2) $O_\$$ *:outputs truly uniform random samples from* $\mathbb{Z}_q^{n\times m} \times \mathbb{Z}_q^{m\times m}$

3 Function Encryption for IPE

3.1 Define

A function encryption for IPE scheme is a series of algorithms (**KeyGen, Enc, Dec**):

KeyGen(λ,ℓ,$\boldsymbol{v}$): On input a security parameter λ, a parameter ℓ represents the length of predicate v and a vector w. The algorithm output public key pk and secret key SK.

Enc($\boldsymbol{M}$,pk,$\boldsymbol{w}$): On input pk, a message matrix $\boldsymbol{M}$ and an attribute vector $\boldsymbol{w}$, the algorithm output ciphertexts C.

Dec(SK,$\boldsymbol{C}$): On input SK and ciphertexts $\boldsymbol{C}$, the algorithm outputs a message $\boldsymbol{M}$.

Correctness: If $< \boldsymbol{v}, \boldsymbol{w} >= 0$, we have:

$$\mathbf{Dec}(\ SK, \mathbf{Enc}(pk, \boldsymbol{M}, \boldsymbol{w}\ )) = \boldsymbol{M}.$$

3.2 Security Model

In this section, we introduced the security model of IPE, which bases on indistinguishability of ciphertexts under chosen-plaintext attack (IND-CPA) game. It means that the challenge ciphertext and random elements in the ciphertext space are indistinguishable.

Init: The adversary $\mathcal{A}$ choose a target attribute vector $\boldsymbol{w}^* = (w_1{}^*, \cdots, w_\ell{}^*) \in \mathbb{Z}_q^\ell$and sends the message to the challenger $\mathcal{C}$.

Setup: The $\mathcal{C}$ accept $\boldsymbol{w}^*$ and $\boldsymbol{M}^*$ from adversary $\mathcal{A}$. The $\mathcal{C}$ runs **Setup** (λ,ℓ) and gives $\mathcal{A}$ the public parameters pp. $\mathcal{C}$ keep the master secret key MSK to itself.

Phase 1: $\mathcal{C}$ answers the $\mathcal{A}$'s private key generation queries on predicate vector $\boldsymbol{v} = (v_1, \cdots, v_\ell) \in \mathbb{Z}_q^\ell$, where $< \boldsymbol{v}, \boldsymbol{w}^* >\neq 0$. In order to get the private key, $\mathcal{C}$ runs **KeyGen**(pp,MSK,$\boldsymbol{v}$) and then $\mathcal{C}$ sends the algorithm output result to the $\mathcal{A}$.

Challenge: After the $\mathcal{A}$ finishes the Phase 1, and $\mathcal{A}$ outputs a challenge plaintext $\boldsymbol{M}^*$ to $\mathcal{C}$. Therefore, $\mathcal{C}$ selects a bit $b \leftarrow \{0,1\}$ randomly. If $b = 0$, $\mathcal{C}$ sends $CT=$ **Enc** (pp,$\boldsymbol{w}^*$,b) to $\mathcal{A}$. If $b = 1$, $\mathcal{C}$ chooses a random ciphertext $CT = CT^*$ from the ciphertext space and sends it to the $\mathcal{A}$.

Phase 2: $\mathcal{A}$ issues additional private key generation queries on the predicate vector where $< \boldsymbol{v}, \boldsymbol{w}^* >\neq 0$. The $\mathcal{C}$ responds as in Phase 1.

Guess: Finally, $\mathcal{A}$ outputs a guess $b' \in \{0,1\}$. If $b = b'$, the $\mathcal{A}$ wins the game.

Definition 6. *We define the advantage of the* $\mathcal{A}$ *as follows:*

$$Adv_{\mathcal{A}}(\lambda) = |\Pr[b = b'] - 1/2|.$$

4 Multi-bit Function Encryption for IPE

4.1 Construction

Setup(λ,ℓ): A security parameter λ and a parameter ℓ represents the length of predicate vector and attribute vector as input. Next do:

1) Generate a uniformly random matrix $\boldsymbol{A} \in \mathbb{Z}_q^{n\times m}$ and a short basis $\boldsymbol{T_A} \in \mathbb{Z}^{m\times m}$ for $\Lambda_q^\perp(\boldsymbol{A})$ using $TrapGen(q, n, m)$.
2) Select $\ell \cdot (1 + k)$ uniformly random a matrix $\boldsymbol{U}_{i,\xi} \in \mathbb{Z}_q^{n\times m}$ and a matrix $\boldsymbol{U}_0 \in \mathbb{Z}_q^{n\times m}$,where $i = 1, \cdots, \ell$ and $\xi = 0, \cdots, k$.
3) Output the pp=$(\boldsymbol{A}, \{\boldsymbol{U}_{i,\xi}\}_{i\in\{1,\cdots,\ell\},\xi\in\{0,\cdots,k\}}, \boldsymbol{U}_0)$ and the MSK=$\boldsymbol{T_A}$.

KeyGen(pp,MSK,$\boldsymbol{v}$): the pp, a MSK, and a vector $\boldsymbol{v} = (v_1, \cdots, v_\ell) \in \mathbb{Z}_q^\ell$ as input. Next do:

1) For $i = 1, \cdots, \ell$, write the r-ary decomposition of v_i as

$$v_i = \sum_{\xi=0}^{k} v_{i,\xi} \cdot r^\xi$$

where $v_{i,\xi}$ are integers in $[0,\ r-1]$.
2) Let the matrix

$$\boldsymbol{U} = \boldsymbol{U}_0 + \sum_{i=1}^{\ell}\sum_{\xi=0}^{k} v_{i,\xi}\boldsymbol{U}_{i,\xi} = (\boldsymbol{u}_1, \cdots, \boldsymbol{u}_m)$$

where $\boldsymbol{u}_i \in \mathbb{Z}_q^n$.
3) Sample $\boldsymbol{e}_i \leftarrow Sample\Pr e(\boldsymbol{A}, \boldsymbol{T_A}, \boldsymbol{u}_i, \sigma)$, where $\boldsymbol{e}_i \in \mathbb{Z}_q^m$ and $\boldsymbol{A} \cdot \boldsymbol{e}_i = \boldsymbol{u}_i \bmod q$. Let $\boldsymbol{E} = (\boldsymbol{e}_1, \cdots \boldsymbol{e}_m)$, then $\boldsymbol{A} \cdot \boldsymbol{E} = \boldsymbol{U} \bmod q$.
4) Output $SK = \boldsymbol{E}$.

Enc(pp,$\boldsymbol{w}$,$\boldsymbol{M}$): The pp, and a vector $\boldsymbol{w} = (w_1, w_2, \cdots, w_l) \in \mathbb{Z}_q^l$, and a message plaintext vector $\boldsymbol{M} \in \{0,1\}^{m\times m}$ as input, do:

1) Select a uniformly random matrix $\boldsymbol{B} \leftarrow \mathbb{Z}_q^{n\times m}$ and a matrix $\boldsymbol{S} \leftarrow \mathbb{Z}_q^{n\times m}$.
2) Select a noise matrix $\boldsymbol{X} \leftarrow \bar{\Psi}_\alpha^{m\times m}$.
3) Let $\boldsymbol{C}_0 = \boldsymbol{A}^{\mathrm{T}}\boldsymbol{S} + 2\boldsymbol{X}$, $\boldsymbol{C}' = \boldsymbol{U}_0{}^{\mathrm{T}}\boldsymbol{S} + 2\boldsymbol{X} + \boldsymbol{M}$ and $\boldsymbol{C}_{i,\xi} = (\boldsymbol{U}_{i,\xi} + r^{\xi}\boldsymbol{w}_i\boldsymbol{B})^{\mathrm{T}}\boldsymbol{S} + 2\boldsymbol{R}_{i,\xi}{}^{\mathrm{T}}\boldsymbol{X}$, where $\boldsymbol{R}_{i,\xi} \in \{-1,1\}^{m\times m}$, $i = 1,\cdots,\ell$ and $\xi = 0\cdots,k$.
4) Finally,the ciphertext CT= $(\boldsymbol{C}_0,\{\boldsymbol{C}_{i,\xi}\}_{i\in\{1,\cdots,\ell\},\xi\in\{0,\cdots,k\}},\boldsymbol{C}')$ is the output.

Dec(pp,SK,CT): The pp, a SK, and CT as input, do:

1) Compute $\boldsymbol{C}_{\boldsymbol{v}} = \sum_{i=1}^{\ell}\sum_{\xi=0}^{k} v_{i,\xi}\boldsymbol{C}_{i,\xi}$
2) Let $\boldsymbol{Y} = \boldsymbol{C}' + \boldsymbol{C}_{\boldsymbol{v}}$.
3) Compute $\boldsymbol{M} = \boldsymbol{Y} - \boldsymbol{E}^{\mathrm{T}}\boldsymbol{C}_0(\mathrm{mod}2)$.

4.2 Correctness

The calculation for decryption is as follows:

$$\begin{aligned}\boldsymbol{C}_{\boldsymbol{v}} &= \sum_{i=1}^{\ell}\sum_{\xi=0}^{k} v_{i,\xi}\boldsymbol{C}_{i,\xi}\\ &= \sum_{i=1}^{\ell}\sum_{\xi=0}^{k} v_{i,\xi}((\boldsymbol{U}_{i,\xi} + r^{\xi}w_i\boldsymbol{B})^{\mathrm{T}}\boldsymbol{S} + 2\boldsymbol{R}_{i,\xi}{}^{\mathrm{T}}\boldsymbol{X})\\ &= \sum_{i=1}^{\ell}\sum_{\xi=0}^{k}(v_{i,\xi}\boldsymbol{U}_{i,\xi})^{\mathrm{T}}\boldsymbol{S} + \underbrace{(\sum_{i=1}^{\ell}\sum_{\xi=0}^{k} r^{\xi}v_{i,\xi}w_i)}_{<\boldsymbol{v},\boldsymbol{w}>}\boldsymbol{B}^{\mathrm{T}}\boldsymbol{S} +\\ &\quad 2\sum_{i=1}^{\ell}\sum_{\xi=0}^{k} v_{i,\xi}\boldsymbol{R}_{i,\xi}{}^{\mathrm{T}}\boldsymbol{X}\end{aligned}$$

If the user meets the decryption requirements, that is,$<\boldsymbol{v},\boldsymbol{w}>= 0(\mathrm{mod}q)$, let:

$$\boldsymbol{C}_{\boldsymbol{v}} = \sum_{i=1}^{\ell}\sum_{\xi=0}^{k}(v_{i,\xi}\boldsymbol{U}_{i,\xi})^{\mathrm{T}}\boldsymbol{S} + 2\sum_{i=1}^{\ell}\sum_{\xi=0}^{k} v_{i,\xi}\boldsymbol{R}_{i,\xi}{}^{\mathrm{T}}\boldsymbol{X}$$

And then:

$$\begin{aligned}\boldsymbol{Y} &= \boldsymbol{C}' + \boldsymbol{C}_{\boldsymbol{v}}\\ &= \boldsymbol{U}_0{}^{\mathrm{T}}\boldsymbol{S} + 2\boldsymbol{X} + \boldsymbol{M} + \sum_{i=1}^{\ell}\sum_{\xi=0}^{k}(v_{i,\xi}\boldsymbol{U}_{i,\xi})^{\mathrm{T}}\boldsymbol{S}\\ &\quad + 2\sum_{i=1}^{\ell}\sum_{\xi=0}^{k} v_{i,\xi}\boldsymbol{R}_{i,\xi}{}^{\mathrm{T}}\boldsymbol{X}\\ &= (\boldsymbol{U}_0 + \sum_{i=1}^{\ell}\sum_{\xi=0}^{k} v_{i,\xi}\boldsymbol{U}_{i,\xi})^{\mathrm{T}}\boldsymbol{S} + 2\boldsymbol{X} + \boldsymbol{M}\\ &\quad + 2\sum_{i=1}^{\ell}\sum_{\xi=0}^{k} v_{i,\xi}\boldsymbol{R}_{i,\xi}{}^{\mathrm{T}}\boldsymbol{X}\\ &= \boldsymbol{U}^{\mathrm{T}}\boldsymbol{S} + 2\boldsymbol{X} + \boldsymbol{M} + 2\sum_{i=1}^{\ell}\sum_{\xi=0}^{k} v_{i,\xi}\boldsymbol{R}_{i,\xi}{}^{\mathrm{T}}\boldsymbol{X}\end{aligned}$$

Finally, we compute:

$$\begin{aligned}&\boldsymbol{Y} - \boldsymbol{E}^{\mathrm{T}}\boldsymbol{C}_0\\ &= [\boldsymbol{U}^{\mathrm{T}}\boldsymbol{S} + 2\boldsymbol{X} + \boldsymbol{M} + 2\sum_{i=1}^{\ell}\sum_{\xi=0}^{k} v_{i,\xi}\boldsymbol{R}_{i,\xi}{}^{\mathrm{T}}\boldsymbol{X}\\ &\quad -\boldsymbol{E}^{\mathrm{T}}(\boldsymbol{A}^{\mathrm{T}}\boldsymbol{S} + 2\boldsymbol{X})](\mathrm{mod}2)\\ &= [\boldsymbol{U}^{\mathrm{T}}\boldsymbol{S} + 2\boldsymbol{X} + \boldsymbol{M} + 2\sum_{i=1}^{\ell}\sum_{\xi=0}^{k} v_{i,\xi}\boldsymbol{R}_{i,\xi}{}^{\mathrm{T}}\boldsymbol{X}\\ &\quad -(\boldsymbol{AE})^{\mathrm{T}}\boldsymbol{S} - 2\boldsymbol{E}^{\mathrm{T}}\boldsymbol{X}](\mathrm{mod}\ 2)\\ &= (\boldsymbol{M} + 2\boldsymbol{X} - 2\boldsymbol{E}^{\mathrm{T}}\boldsymbol{X} + 2\sum_{i=1}^{\ell}\sum_{\xi=0}^{k} v_{i,\xi}\boldsymbol{R}_{i,\xi}{}^{\mathrm{T}}\boldsymbol{X})(\mathrm{mod}\ 2)\\ &= \boldsymbol{M}\ \mathrm{mod}\ 2\\ &= \boldsymbol{M}\end{aligned}$$

4.3 Security Analysis

Theorem 1. *Under the assumption of the LWE hard problem, our scheme is IND-CPA security in the standard model [2].*

Proof. In the process of proving our scheme, some games are needed to show that the advantage of the polynomial adversary is negligible.

Game 0: This is the original IND-CPA game between an adversary $\mathcal{A}$ against our scheme and an IND-CPA challenger $\mathcal{C}$.

Game 1: The difference between Game 0 and Game 1 is the choice of matrix $\boldsymbol{A}$ and $\boldsymbol{U}_{i,\xi}$. In **Game 0**, the challenger $\mathcal{C}$ generates $\boldsymbol{A}$ in $\mathbb{Z}_q^{n\times m}$ by running $TrapGen(q,n,m)$. According to **Lemma 1**, the matrix $\boldsymbol{A}$ is statistically close to a uniformly random matrix. In **Game 1**, we choose a matrix $\boldsymbol{A}$ randomly in $\mathbb{Z}_q^{n\times m}$. Obviously, the matrix $\boldsymbol{A}$ is indistinguishable from **Game 0**.

Let $\boldsymbol{R}^*{}_{i,\xi} \in \{-1,1\}^{m\times m}$ is selected from random matrix, and $\boldsymbol{B}^*$ is chosen from a random matrix in $\mathbb{Z}_q^{n\times m}$ and an attribute vector $\boldsymbol{w}^* = (w_1{}^*,\cdots,w_\ell{}^*) \in \mathbb{Z}_q^{\ell}$ that $\mathcal{A}$ intends to attack. $\mathcal{C}$ constructs $\boldsymbol{U}_{i,\xi}$ as $\boldsymbol{U}_{i,\xi} \leftarrow 2\boldsymbol{A}\boldsymbol{R}_{i,\xi}{}^* - r^{\xi}w_i{}^*\boldsymbol{B}^*$. So, for $i \in \{1,\cdots,\ell\}$ and $\xi \in \{0\cdots,k\}$, the matrix $\boldsymbol{U}_{i,\xi}$ is statistically close to a uniformly random matrix in $\mathbb{Z}_q^{n\times m}$.

In **Game 0** the matrix $\boldsymbol{U}_{i,\xi}$ is uniformly random in $\mathbb{Z}_q^{n\times m}$. From the above analysis, the matrix $\boldsymbol{U}_{i,\xi}$ in **Game 0** and the matrix $\boldsymbol{U}_{i,\xi}$ in **Game 1** are not distinguishable.

Game 2: The difference between **Game 1** and **Game 2** is the generating method of matrix $\boldsymbol{E}$,$\boldsymbol{U}_0$. The choice of $\boldsymbol{A}$ remains as in **Game 1**. To respond to the $\mathcal{A}$'s private key queries of a vector $\boldsymbol{w}^*$, where $<\boldsymbol{v},\boldsymbol{w}^*>\neq 0$. And the $\mathcal{C}$ needs a matrix $\boldsymbol{E}$.Firstly, $\mathcal{C}$ randomly select matrix $\boldsymbol{E}$ according to Gauss distribution $D_{s,0}$.Then, $\mathcal{C}$ compute $\boldsymbol{A}\cdot\boldsymbol{E} = \boldsymbol{U}\ \mathrm{mod}\ q$. Because $\boldsymbol{E}$ is selected from the Gauss distribution and $\boldsymbol{A}$ is a random matrix, the matrix $\boldsymbol{U}$ is statistically indistinguishable from a uniformly random matrix.

Let $\boldsymbol{U} = \boldsymbol{U}_0 + \sum_{i=1}^{\ell}\sum_{\xi=0}^{k} v_{i,\xi}\boldsymbol{U}_{i,\xi}$, where $\boldsymbol{U}_{i,\xi}$ remains as in **Game 1** and $v_{i,\xi}$ as in **KeyGen**. Given $\boldsymbol{U}, \boldsymbol{U}_{i,\xi}, v_{i,\xi}$, we can calculate $\boldsymbol{U}_0$. From the above analysis,$\boldsymbol{U}$,$\boldsymbol{U}_{i,\xi} \in \mathbb{Z}_q^{n\times m}$ are statistically close to uniformly random. Yet, $\boldsymbol{U}_0$ can be regarded as a uniformly random matrix. So, the matrix $\boldsymbol{U}_0$ is not distinguishable from **Game 1**.

Game 3: The challenge ciphertext triples ($\boldsymbol{C}_0$, $\{\boldsymbol{C}_{i,\xi}\}_{i\in\{1,\cdots,\ell\},\xi\in\{0,\cdots,k\}}, \boldsymbol{C}'$) is chosen as random independent elements in $\mathbb{Z}_q^m$. Because the challenge ciphertext is always random elements, the adversary's advantage is zero. Other parts are the same as **Game 2**.

For a PPT adversary, we will show that computationally indistinguishable between **Game 1** and **Game 2**. Then, we reduce it to the LWE hard problem.

Reduction from LWE: In distinguishing between **Game 1** and **Game 2**, we assume that the $\mathcal{A}$ has non-negligible advantage. Therefore, we construct an LWE algorithm $\mathcal{B}$.

Instance. $\mathcal{B}$ has received m pairs random instances $(\boldsymbol{A}, \boldsymbol{A}^{\mathrm{T}}\boldsymbol{S} + \boldsymbol{X}) \in \mathbb{Z}_q^{n\times m} \times \mathbb{Z}_q^{m\times m}$ from O.

Targeting. The $\mathcal{A}$ claims to $\mathcal{B}$ an attribute vector $\boldsymbol{w}^*$ that it is going to be attacked.

Setup. The $\mathcal{B}$ constructs the pp as follows:

1) We let $\boldsymbol{A} \in \mathbb{Z}_q^{n\times m}$ as one of the pp.
2) The matrix $\boldsymbol{U}_0$ and $\boldsymbol{U}_{i,\xi}$ as mentioned in **Game 2**, where $i \in \{1,\cdots,\ell\}$ and $\xi \in \{0\cdots,k\}$.

Queries. For $\mathcal{A}$'s the private key generation query of a predicate vector $\boldsymbol{v}$ ($< \boldsymbol{v}, \boldsymbol{w}^* > \neq 0$), $\mathcal{B}$'s answer is as **Game 2**.

Challenge. The $\mathcal{B}$ response to a challenge ciphertext for the target attribute vector $\boldsymbol{w}^*$, and $\mathcal{A}$ received a message $\boldsymbol{M}^* \in \{0,1\}^{m\times m}$, as follows:

1) $\boldsymbol{C}_0 = 2\boldsymbol{C} = 2\boldsymbol{A}^{\mathrm{T}}\boldsymbol{S} + 2\boldsymbol{X} = \boldsymbol{A}^{\mathrm{T}}\boldsymbol{S}' + 2\boldsymbol{X}$, where $\boldsymbol{S}' = 2\boldsymbol{S}$
2) $\boldsymbol{C}' = {\boldsymbol{U}_0}^{\mathrm{T}}\boldsymbol{S} + 2\boldsymbol{X} + \boldsymbol{M}^*$
3) $\boldsymbol{C}_{i,\xi} = (\boldsymbol{U}_{i,\xi} + r^{\xi}{w_i}^*\boldsymbol{B})^{\mathrm{T}}\boldsymbol{S} + 2{\boldsymbol{R}^*}_{i,\xi}^{\mathrm{T}}\boldsymbol{X}$ $= (2\boldsymbol{A}{\boldsymbol{R}_{i,\xi}}^* - r^{\xi}{w_i}^*\boldsymbol{B}^* + r^{\xi}{w_i}^*\boldsymbol{B}^*)^{\mathrm{T}}\boldsymbol{S} + 2{\boldsymbol{R}^*}_{i,\xi}^{\mathrm{T}}\boldsymbol{X}$ where $i \in \{1,\cdots,\ell\}$ and $\xi \in \{0\cdots,k\}$. Then $(\boldsymbol{C}_0, \{\boldsymbol{C}_{i,\xi}\}_{i\in\{1,\cdots,\ell\},\xi\in\{0,\cdots,k\}}, \boldsymbol{C}')$ as a challenge ciphertext.
4) Randomly choose a bit $b \leftarrow \{0,1\}$. If $b = 0$ send $CT = (\boldsymbol{C}_0, \{\boldsymbol{C}_{i,\xi}\}_{i\in\{1,\cdots,\ell\},\xi\in\{0,\cdots,k\}}, \boldsymbol{C}')$ to the adversary $\mathcal{A}$. If $b = 1$ choose a random $(\boldsymbol{C}_0, \{\boldsymbol{C}_{i,\xi}\}_{i\in\{1,\cdots,\ell\},\xi\in\{0,\cdots,k\}}, \boldsymbol{C}') \in \mathbb{Z}_q^{m\times m}$ from the ciphertext space, and send it to the $\mathcal{A}$. When the LWE oracle is pseudo-random (i.e.O=O_s), the distribution of the challenge ciphertext is the same as in **Game 2**. When O=O_s, the challenge ciphertext is chosen from **Game 2**.

Guess. When $\mathcal{A}$ guesses a right b, $\mathcal{B}$ outputs 1, otherwise $\mathcal{B}$ outputs 0.

Therefore, $\mathcal{B}$'s advantage in solving LWE hard problem is the advantage of the $\mathcal{A}$ to distinguish between **Game 2** and **Game 3**. If $\mathcal{A}$ can distinguish the above games, then $\mathcal{B}$ can solve LWE problem.

□

4.4 Comparison

In this summary, we compare our scheme with other schemes [25, 27] in terms of private key size, attribute-hiding characteristics, and bit encryption. Table 1 is the comparison result.

5 Extension

5.1 Construction

Setup(λ,ℓ): A security parameter λ and a parameter ℓ represents the length of predicate vector and attribute vector as input. Next do:

1) Generate a uniformly random matrix $\boldsymbol{A} \in \mathbb{Z}_q^{n\times m}$ and short basis $\boldsymbol{T}_{\boldsymbol{A}} \in \mathbb{Z}^{m\times m}$ for $\Lambda_q^{\perp}(\boldsymbol{A})$ using $TrapGen(q,n,m)$.
2) Select $\ell\cdot(1+k)$ uniformly random a matrix $\boldsymbol{U}_{i,\xi} \in \mathbb{Z}_q^{n\times m}$ and a matrix $\boldsymbol{U}_0 \in \mathbb{Z}_q^{n\times m}$,where $i = 1,\cdots,\ell$ and $\xi = 0,\cdots,k$.
3) Output the pp=$(\boldsymbol{A}, \{\boldsymbol{U}_{i,\xi}\}_{i\in\{1,\cdots,\ell\},\xi\in\{0,\cdots,k\}}, \boldsymbol{U}_0)$ and the MSK=$\boldsymbol{T}_{\boldsymbol{A}}$.

KeyGen(pp,MSK,$\boldsymbol{v}$): The pp, a MSK, and a vector $\boldsymbol{v} = (v_1,\cdots,v_\ell) \in_q^\ell$ as input. Next do:

1) For $i = 1,\cdots,\ell$, write the r-ary decomposition of v_i as
$$v_i = \sum_{\xi=0}^{k} v_{i,\xi}\cdot r^{\xi}$$
where $v_{i,\xi}$ are integers in $[0, r-1]$.
2) Let the matrix
$$\boldsymbol{U} = \boldsymbol{U}_0 + \sum_{i=1}^{\ell}\sum_{\xi=0}^{k} v_{i,\xi}\boldsymbol{U}_{i,\xi} = (\boldsymbol{u}_1,\cdots,\boldsymbol{u}_m)$$
where $\boldsymbol{u}_i \in \mathbb{Z}_q^n$.
3) Sample $\boldsymbol{e}_i \leftarrow Sample\Pr e(\boldsymbol{A}, \boldsymbol{T}_{\boldsymbol{A}}, \boldsymbol{u}_i, \sigma)$, where $\boldsymbol{e}_i \in \mathbb{Z}_q^m$ and $\boldsymbol{A}\cdot\boldsymbol{e}_i = \boldsymbol{u}_i \bmod q$. Let $\boldsymbol{E} = (\boldsymbol{e}_1,\cdots\boldsymbol{e}_m)$, then $\boldsymbol{A}\cdot\boldsymbol{E} = \boldsymbol{U} \bmod q$.
4) Output $SK = \boldsymbol{E}$.

Enc(pp,$\boldsymbol{w}$,$\boldsymbol{M}$): The pp, and a vector $\boldsymbol{w} = (w_1, w_2,\cdots,w_l) \in \mathbb{Z}_q^l$, and a message plaintext vector $\boldsymbol{M} \in \mathbb{Z}_p^{m\times m}$ as input, do:

Table 1: Comparison of related literature

Cryptosystem	Private key size	Attribute-hiding	Bit encryption	Security Assumption
[27]	$o(mn\log q)$	No	Single-bit	Bilinear Pairing
[25]	$o(mn\log q)$	No	Single-bit	LWE
Our scheme	$o(mn\log q)$	Yes	Multi-bit Matrix	LWE

1) Select a uniformly random matrix $\boldsymbol{B} \leftarrow \mathbb{Z}_q^{n\times m}$ and a matrix $\boldsymbol{S} \leftarrow \mathbb{Z}_q^{n\times m}$.

2) Select a noise matrix $\boldsymbol{X} \leftarrow \bar{\Psi}_\alpha^{m\times m}$.

3) Let $\boldsymbol{C}_0 = \boldsymbol{A}^{\mathrm{T}}\boldsymbol{S} + p\boldsymbol{X}$, $\boldsymbol{C}' = \boldsymbol{U}_0{}^{\mathrm{T}}\boldsymbol{S} + p\boldsymbol{X} + \boldsymbol{M}$ and $\boldsymbol{C}_{i,\xi} = (\boldsymbol{U}_{i,\xi} + r^{\xi}\boldsymbol{w}_i\boldsymbol{B})^{\mathrm{T}}S + p\boldsymbol{R}_{i,\xi}{}^{\mathrm{T}}\boldsymbol{X}$, where $\boldsymbol{R}_{i,\xi} \in \{-1,1\}^{m\times m}$, $i = 1,\cdots,\ell$ and $\xi = 0,\cdots,k$.

4) Finally,the ciphertext CT= $(\boldsymbol{C}_0,\{\boldsymbol{C}_{i,\xi}\}_{i\in\{1,\cdots,\ell\},\xi\in\{0,\cdots,k\}},\boldsymbol{C}')$ is the output.

Dec(pp,SK,CT): The pp, a SK, and CT as input, do:

1) Compute $\boldsymbol{C}_{\boldsymbol{v}} = \sum_{i=1}^{\ell}\sum_{\xi=0}^{k} v_{i,\xi}\boldsymbol{C}_{i,\xi}$

2) Let $\boldsymbol{Y} = \boldsymbol{C}' + \boldsymbol{C}_{\boldsymbol{v}}$.

3) Compute $\boldsymbol{M} = \boldsymbol{Y} - \boldsymbol{E}^{\mathrm{T}}\boldsymbol{C}_0(\mathrm{mod}\ p)$.

5.2 Correctness

The calculation for decryption is as follows:

$$\begin{aligned}\boldsymbol{C}_{\boldsymbol{v}} &= \sum_{i=1}^{\ell}\sum_{\xi=0}^{k} v_{i,\xi}\boldsymbol{C}_{i,\xi}\\ &= \sum_{i=1}^{\ell}\sum_{\xi=0}^{k} v_{i,\xi}((\boldsymbol{U}_{i,\xi} + r^{\xi}w_i\boldsymbol{B})^{\mathrm{T}}\boldsymbol{S} + p\boldsymbol{R}_{i,\xi}{}^{\mathrm{T}}\boldsymbol{X})\\ &= \sum_{i=1}^{\ell}\sum_{\xi=0}^{k}(v_{i,\xi}\boldsymbol{U}_{i,\xi})^{\mathrm{T}}\boldsymbol{S} + \underbrace{(\sum_{i=1}^{\ell}\sum_{\xi=0}^{k} r^{\xi}v_{i,\xi}w_i)}_{<\boldsymbol{v},\boldsymbol{w}>}\boldsymbol{B}^{\mathrm{T}}\boldsymbol{S} +\\ &\quad p\sum_{i=1}^{\ell}\sum_{\xi=0}^{k} v_{i,\xi}\boldsymbol{R}_{i,\xi}{}^{\mathrm{T}}\boldsymbol{X}.\end{aligned}$$

If the user meets the decryption requirements, that is,$<\boldsymbol{v},\boldsymbol{w}>= 0(\mathrm{mod}q)$, let:

$$\boldsymbol{C}_{\boldsymbol{v}} = \sum_{i=1}^{\ell}\sum_{\xi=0}^{k}(v_{i,\xi}\boldsymbol{U}_{i,\xi})^{\mathrm{T}}\boldsymbol{S} + p\sum_{i=1}^{\ell}\sum_{\xi=0}^{k} v_{i,\xi}\boldsymbol{R}_{i,\xi}{}^{\mathrm{T}}\boldsymbol{X}.$$

And then:

$$\begin{aligned}\boldsymbol{Y} &= \boldsymbol{C}' + \boldsymbol{C}_{\boldsymbol{v}}\\ &= \boldsymbol{U}_0{}^{\mathrm{T}}\boldsymbol{S} + p\boldsymbol{X} + \boldsymbol{M} + \sum_{i=1}^{\ell}\sum_{\xi=0}^{k}(v_{i,\xi}\boldsymbol{U}_{i,\xi})^{\mathrm{T}}\boldsymbol{S}\\ &\quad + p\sum_{i=1}^{\ell}\sum_{\xi=0}^{k} v_{i,\xi}\boldsymbol{R}_{i,\xi}{}^{\mathrm{T}}\boldsymbol{X}\\ &= (\boldsymbol{U}_0 + \sum_{i=1}^{\ell}\sum_{\xi=0}^{k} v_{i,\xi}\boldsymbol{U}_{i,\xi})^{\mathrm{T}}\boldsymbol{S} + p\boldsymbol{X} + \boldsymbol{M}\\ &\quad + p\sum_{i=1}^{\ell}\sum_{\xi=0}^{k} v_{i,\xi}\boldsymbol{R}_{i,\xi}{}^{\mathrm{T}}\boldsymbol{X}\\ &= \boldsymbol{U}^{\mathrm{T}}\boldsymbol{S} + p\boldsymbol{X} + \boldsymbol{M} + p\sum_{i=1}^{\ell}\sum_{\xi=0}^{k} v_{i,\xi}\boldsymbol{R}_{i,\xi}{}^{\mathrm{T}}\boldsymbol{X}.\end{aligned}$$

Finally, we compute:

$$\begin{aligned}&\boldsymbol{Y} - \boldsymbol{E}^{\mathrm{T}}\boldsymbol{C}_0\\ &= [\boldsymbol{U}^{\mathrm{T}}\boldsymbol{S} + p\boldsymbol{X} + \boldsymbol{M} + p\sum_{i=1}^{\ell}\sum_{\xi=0}^{k} v_{i,\xi}\boldsymbol{R}_{i,\xi}{}^{\mathrm{T}}\boldsymbol{X}\\ &\quad -\boldsymbol{E}^{\mathrm{T}}(\boldsymbol{A}^{\mathrm{T}}\boldsymbol{S} + p\boldsymbol{X})](\mathrm{mod}p)\\ &= [\boldsymbol{U}^{\mathrm{T}}\boldsymbol{S} + p\boldsymbol{X} + \boldsymbol{M} + p\sum_{i=1}^{\ell}\sum_{\xi=0}^{k} v_{i,\xi}\boldsymbol{R}_{i,\xi}{}^{\mathrm{T}}\boldsymbol{X}\\ &\quad -(\boldsymbol{AE})^{\mathrm{T}}\boldsymbol{S} - p\boldsymbol{E}^{\mathrm{T}}\boldsymbol{X}](\mathrm{mod}p)\\ &= (\boldsymbol{M} + p\boldsymbol{X} - p\boldsymbol{E}^{\mathrm{T}}\boldsymbol{X} + p\sum_{i=1}^{\ell}\sum_{\xi=0}^{k} v_{i,\xi}\boldsymbol{R}_{i,\xi}{}^{\mathrm{T}}\boldsymbol{X})(\mathrm{mod}p)\\ &= \boldsymbol{M}\ \mathrm{mod}\ p\\ &= \boldsymbol{M}.\end{aligned}$$

5.3 Security

In this part, it is roughly the same as 4.3, so we do not describe it in detail. The only difference is the message plaintext. In 5.1, the ciphertext on $\mathbb{Z}_p^{m\times m}$ is encrypted. So, in 5.2, we only need to change $\boldsymbol{U}_{i,\xi} \leftarrow 2\boldsymbol{A}\boldsymbol{R}_{i,\xi}{}^{*} - r^{\xi}w_i{}^{*}\boldsymbol{B}^{*}$ in **Game 2** to $\boldsymbol{U}_{i,\xi} \leftarrow p\boldsymbol{A}\boldsymbol{R}_{i,\xi}{}^{*} - r^{\xi}w_i{}^{*}\boldsymbol{B}^{*}$, and **change** the 2 in the ciphertext triples of Challenge in **Reduction from LWE** to p.

6 Conclusion

Due to the various privacy concerns in the Internet and the inefficiency of known functional encryption schemes. Therefore, in this paper, we propose a high-efficiency inner product predicate scheme on the grid to securely share data in the Internet. In this paper, we use the BGN-type Cryptosystem in [9] to propose a lattice-based multi-bit matrix IPE encryption scheme, and then this greatly expands the plaintext space and improves the efficiency of

encryption. Moreover, under the standard model, our scheme is CPA secure and can resist quantum attacks.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (61902140, 61802110), the Anhui Provincial Natural Science Foundation (1908085QF288), and in part by the Natural Science Foundation of Anhui University (KJ2020A0032), the Nature Science Foundation of Anhui Higher Education Institutions (KJ2021A0527).

References

[1] E. Affum, X. S. Zhang, and X. F. Wang, "Lattice cp-abe scheme supporting reduced-obdd structure," in *Advances in Computer, Communication and Computational Sciences*, pp. 131–142, 2021.

[2] S. Agrawal, D. Boneh, and X. Boyen, "Efficient lattice (h) ibe in the standard model," in *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 553–572, 2010.

[3] S. Agrawal, D. M. Freeman, and V. Vaikuntanathan, "Functional encryption for inner product predicates from learning with errors," in *International Conference on the Theory and Application of Cryptology and Information Security*, pp. 21–40, 2011.

[4] S. Agrawal, M. Maitra, and S. Yamada, "Attribute based encryption (and more) for nondeterministic finite automata from lwe," in *Annual International Cryptology Conference*, pp. 765–797, 2019.

[5] M. Ali., M. Sadeghi, and X. M. Liu, "Lightweight revocable hierarchical attribute-based encryption for internet of things," *IEEE Access*, vol. 8, pp. 23951–23964, 2020.

[6] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attribute-based encryption," in *IEEE symposium on security and privacy (SP'07)*, pp. 321–334, 2007.

[7] D. Boneh, A. Sahai, and B. Waters, "Functional encryption: Definitions and challenges," in *Theory of Cryptography Conference*, pp. 253–273, 2011.

[8] S. Chatterjee and S. Mukherjee, "Large universe subset predicate encryption based on static assumption (without random oracle)," in *Cryptographers' Track at the RSA Conference*, pp. 62–82, 2019.

[9] C. Gentry, S. Halevi, and V. Vaikuntanathan, "A simple bgn-type cryptosystem from lwe," in *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 506–522, 2010.

[10] C. Gentry, C. Peikert, and V. Vaikuntanathan, "How to use a short basis: Trapdoors for hard lattices and new cryptographic constructions," in *Proceedings of the 40th Annual ACM Symposium on Theory of Computing*, pp. 197–206, 2009.

[11] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data," in *Proceedings of the 13th ACM Conference on Computer and Communications Security*, pp. 89–98, 2006.

[12] J. Howe, A. Khalid, C. Raffertyand, F. Regazzoni, "On practical discrete gaussian samplers for lattice-based cryptography," *IEEE Transactions on Computers*, vol. 67, no. 3, pp. 322–334, 2016.

[13] C. H. Jiao and X. Y. Xiang, "Attribute-based encryption supporting data filtration over post-quantum assumptions," *International Journal of Electronic Security and Digital Forensics*, vol. 10, no. 4, pp. 323–337, 2018.

[14] J. Katz, A. Sahai, and B. Waters, "Predicate encryption supporting disjunctions, polynomial equations, and inner products," in *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 146–162, 2008.

[15] F. Khan, H. Li, Y. H. Zhang, H. Abbas, and T. Yaqoob, "Efficient attribute-based encryption with repeated attributes optimization," *International Journal of Information Security*, vol. 20, no. 3, pp. 431–444, 2021.

[16] L. Liu, Z. Z. Guo, Z. Cao, Z. Chen, "An improvement of one anonymous identity-based encryption scheme," *International Journal of Electronics and Information Engineering*, vol. 9, no. 1, pp. 11–21, 2018.

[17] D. Micciancio and S. Goldwasser, *Complexity of Lattice Problems: A Cryptographic Perspective*, New York, NY, USA: Springer Science & Business Media, 2002.

[18] S. Patranabis, D. Mukhopadhyay, S. C. Ramanna, "Function private predicate encryption for low min-entropy predicates," in *IACR International Workshop on Public Key Cryptography*, pp. 189–219, 2019.

[19] O. Regev, "On lattices, learning with errors, random linear codes, and cryptography," *Journal of the ACM*, vol. 56, no. 6, pp. 1–40, 2009.

[20] J. Ren, L. Y. Zhang, B. C. Wang, "Decentralized multi-authority attribute-based searchable encryption scheme," *International Journal of Network Security*, vol. 23, no. 2, pp. 332–342, 2021.

[21] A. Sahai and B. Waters, "Fuzzy identity-based encryption," in *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 457–473, 2005.

[22] J. F. Sun, Y. Y. Bao, X. Y. Nie, and H. Xiong, "Attribute-hiding predicate encryption with equality test in cloud computing," *IEEE Access*, vol. 6, pp. 31621–31629, 2018.

[23] Q. T. Tian, D. Z. Han, X. G. Liu, and X. S. Yu, "Lwe-based multi-authority attribute-based encryption scheme with hidden policies," *International Journal of Computational Science and Engineering*, vol. 19, no. 2, pp. 233–241, 2019.

[24] T. van de Kamp, A. Peter, and W. Jonker, "A multi-authority approach to various predicate encryption types," *Designs, Codes and Cryptography*, vol. 88, no. 2, pp. 363–390, 2020.

[25] U. S. Varri, S. K. Pasupuleti, K. V. Kadambari, "CP-ABSEL: Ciphertext-policy attribute-based searchable encryption from lattice in cloud storage," *Peer-to-Peer Networking and Applications*, vol. 14, no. 3, pp. 1290–1302, 2021.

[26] Q. Wu, X. J. Ma, L. Y. Zhang, Y. R. Chen, "Expressive ciphertext policy attribute-based searchable encryption for medical records in cloud," *International Journal of Network Security*, vol. 23, no. 3, pp. 461–472, 2021.

[27] K. Yang, G. Wu, C. Dong, X. Fu, F. Li, T. Wu, "Attribute based encryption with efficient revocation from lattices," *International Journal of Network Security*, vol. 22, no. 1, pp. 161–170, 2020.

Biography

Mingming Jiang Mingming Jiang is a associate professor in the School of Computer Science and Technology, Huaibei Normal University. He received his PhD in cryptography from Xidian University in 2014, and received his MS and BS in cryptography from Huaibei Normal University in 2010 and 2007, respectively. His research interests include public key cryptography based on lattice and provable security.

Qihong Chen MS. of Huaibei Normal University. Her research interests include lattice-based public key cryptography and information security.

Yuyan Guo Associate professor in the School of Computer Science and Technology, Huaibei Normal University. She received her Ph.D. degree in computer science from Hohai University, Nanjing, China in 2016. Her research interests include cryptography and information security.

Dongbing Zhang Born in 1974, Master. Now, he is an associate professor in Huaibei Normal University. His main research interests include algorithm optimization and information security.

Research on Data Hiding Schemes for AMBTC Compressed Images

Kurnia Anggriani[1,2], Nan-I Wu[3], and Min-Shiang Hwang[1,4]
(Corresponding author: Min-Shiang Hwang)

Department of Computer Science & Information Engineering, Asia University[1]
500, Lioufeng Rd., Wufeng, Taichung 41354, Taichung, Taiwan, ROC
Faculty of Engineering, University of Bengkulu, Indonesia[2]
Department of Digital Multimedia, Lee-Ming Institute of Technology[3]
No.2-2, Lijhuan Rd., Taishan Township, Taipei County 243, Taiwan, ROC
Department of Medical Research, China Medical University Hospital, China Medical University, Taiwan[4]
Email: mshwang@asia.edu.tw
(Received Aug. 30, 2022; Revised and Accepted Oct. 20, 2022; First Online Oct. 22, 2022)

Abstract

In this paper, the AMBTC compressed image-based data-hiding research is summarized and introduced to present the current status of the research topic. We detailed explain the embedding procedures and the illustration of each survey paper to compare and evaluate the effectiveness of each method in the surveyed paper. Moreover, we summarize the experimental results and present them in tables and graphs to provide a precise performance comparison. In future work, it is highly desirable to develop a method for optimizing the structure of a compressed AMBTC image to achieve higher hiding capacity and image quality.

Keywords: Absolute Moment Truncation Coding; Compressed Image; Data Hiding; Hamming Distance; Least Significant Bit; Pixel Value Differencing

1 Introduction

Due to the tremendous use of internet technology and application, the need for secure message communication is unavoidable. Moreover, all aspects, such as health care, economics, and political information, are easily exposed on many social media sites such as Facebook and Instagram. Therefore, research on data hiding has attracted many researchers in the recent last year. Data hiding is a private function of concealing secret data in other media with unnoticed changes to human perception. Both secret data and media can be an image, audio, or video type [4, 18].

Data hiding in the compressed image domain has recently become a worthwhile and applicable research topic. It is due to the characteristic of the compressed image, which are low storage, fast transmission, and fast computation [7, 8]. Absolute moment block truncation coding (AMBTC) compressed image has all of the mentioned characteristics [11]. The current AMBTC research topic consists of reversible data hiding (RDH) [2, 3, 6, 9, 10, 13, 25], secret sharing (SS) [15, 19, 22–24], encrypted reversible data hiding (ERDH) [16, 17, 20], and image authentication (IE) [1, 5, 12].

In this paper, we summarized and analyzed some articles in reputable journal papers to present existing methods in the AMBTC RDH method. Furthermore, this survey paper aims to identify a research gap to develop new methods for future research.

To measure the performance of the AMBTC RDH method, there are two main parameters to be concerned. The first is image quality assessment. It compares the quality of an original image and a stego image after being modified with embedded secret bits. The image quality is determined by the peak signal-to-noise ratio (PSNR). Therefore, we must first compute the mean square error (MSE). Equations 1 and 2 provide the PSNR and MSE calculation formulas.

$$PSNR = 10\log_{10}\frac{255^2}{MSE} \quad (1)$$

$$MSE = \frac{1}{M\times M}\sum_{i=1}^{M}\sum_{j=1}^{M}(T_{ij}-T'_{ij})^2 \quad (2)$$

Where T_{ij} represents the pixel location of the image T, located in the i-th row and the j-th column, and T'_{ij} represents a pixel position of the stego image T', located in the i-th row and the j-th column.

The second parameter is hiding capacity evaluation. It calculates the number of secret data bits embedded in the original image. Bits per pixel (bpp) is another hiding capacity evaluation. It means the total number of bits divided by $M \times M$ of the original image size. Undeniably, a

method with a high hiding capacity results in poor image quality. As a result, we require a method that can proportionally balance hiding capacity and image quality.

In this survey paper (SP), we reviewed the five most relatable papers, namely "An Adaptive Reversible Data Hiding Scheme Using AMBTC and Quantization Level Difference" [2] denoted as survey paper 1 (SP1), "Efficient Reversible Data Hiding Scheme for AMBTC-Compressed Images" [13] denoted as survey paper 2 (SP 2), "Steganography Using Quotient Value Differencing and LSB Substitution for AMBTC Compressed Images" [6] denoted as survey paper 3 (SP3), "A Data Hiding Scheme Based on Turtle-shell for AMBTC Compressed Images" [10] denoted as survey paper 4 (SP4) and "Enhanced AMBTC based data hiding method using hamming distance and pixel value differencing" [9] denoted as survey paper 5 (SP5).

In 2021, Chen *et al.* [2] introduced a data hiding scheme based on the quantization level difference (Q_LD) and interpolation technique. This method adaptively conceals the secret messages into bitmap (BM) of absolute moment block truncation coding (AMBTC) compressed images according to the Q_LD between high quantizer (H) and low quantizer (L). This strategy achieves a high embedding capacity as well as good image quality. This paper is signed as SP1.

In 2021, Lin *et al.* [13] proposed a data-hiding method based on the correlation of two mean values (high and low). The goal is to encode secret messages losslessly and create free space for them to be hidden. The experimental results showed that this method achieved a high embedding capacity while ensuring the same PSNR value as the original AMBTC compressed image. This paper is signed as SP2.

In 2020, Horng *et al.* [6] presented an AMBTC-based data hiding that applied Quotient Value Differencing (QVD) and Least Significant Bit (LSB) substitution. By utilizing the data structure of AMBTC compressed image, the secret messages in concealing in the three-phase of embedding. As a result, the embedding capacity of the method performs related works. This paper is signed as SP3.

In 2020, Lee *et al.* [10] investigated a data hiding technique based on a combination of the turtle shell matrix and AMBTC structure. The embedding procedure is divided into three phases: data embedding in quantization levels, data embedding in the bitmap, and data embedding in quantization level order. As a result, this method provided greater embedding capacity and image quality than previous works. This paper is signed as SP4.

In 2019, Kumar *et al.* [9] introduced a data hiding based on hamming distance and pixel value differencing. This method uses two predefined thresholds to classify the block into three categories: Smooth, Less Complex, and Highly Complex. The embedding strategy varies depending on the type of block. As a result, the embedding capacity varies. The experimental results show that this method outperforms related works in capacity and acceptable image quality. This paper is signed as SP5.

Furthermore, this paper will adhere to the following systematics. In Section 2, the related works are presented in detail. Section 3 discusses the comparison of survey papers' performance. Then, in Section 4, future research is provided. Lastly, the paper is concluded in Section 5.

2 Related Works

This section summarizes and analyzes the fifth survey paper (SP1-SP5) by providing a detailed embedding procedure and illustration. We also provide the experimental result of each survey paper.

2.1 Survey Paper 1 (SP1)

Chen *et al.* [2] divide the embedding procedure into three stages. The flowchart of the embedding procedure of Chen *et al.*'s method is shown in Figure 1. The detailed process of Chen *et al.*'s scheme is as follows:

Step 1. AMBTC Encoding
Firstly, manage an AMBTC encoding procedure to obtain high quantizer, low quantizer, and bitmap, H, L, and BM, respectively. Second, compute the Quantization Level Difference (Q_LD) between H and L.

Step 2. Block Classification
Classify the current block into non embeddable or embeddable blocks by using condition. For example, if the difference (d) is between 4 and 63, it is classified as an embeddable block. On the other hand, when d is less than four and greater than 63, the block is classified as non-embedding.

Step 3. Secret Embedding
For the embeddable block, find out the first bitmap value "0" (FL) and the last bitmap value "1", then record the position as r_1 and r_2, respectively. Because of reversibility, we cannot embed a secret message in r_1 and r_2.

For concealing a secret bit into the rest of the bitmap, we first have to calculate the number of bits to embed for each bitmap (a_i) using Equation (3). Then embed the secret bit by using Equation (4). The flowchart of the embedding procedure is presented in Figure 1.

$$a_i = \lfloor \log_2(\frac{d_i}{2}) \rfloor \tag{3}$$

$$P'_n = \begin{cases} h_i, & \text{if } n = r_2 \\ l_i, & \text{if } n = r_1 \\ h_i - M_n, & \text{if } bp_n = 1 \text{ and } n \neq r_1 \text{ or } r_2 \\ l_i + M_n, & \text{if } bp_n = 0 \text{ and } n \neq r_1 \text{ or } r_2 \end{cases} \tag{4}$$

Illustration of SP1.
Suppose we have 4×4 of the size original image and 42 bits of secret messages.

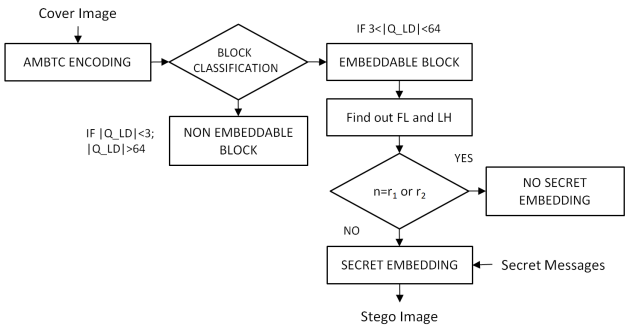


Figure 1: The flowchart of embedding procedure [2]

Step 1. AMBTC Encoding

$$(H, L, BM) = (119, 98, 1000111011001110).$$

Step 2. Block Classification

Since the difference of high and low quantizer $d = 21$ is in the range of 4 and 63. Therefore the block is embeddable.

Step 3. Secret Embedding.

Calculate $a = 3$ using Equation 3, then divide the secret bits into 3 length bits and convert to binary: (101||010||111||010||110||001||011||110||100||000||100||001||101||111).

$$Mn : \{5, N/A, 2, 7, 2, 6, 1, 3, 6, 4, 0, 4, 1, 5, N/A, 7\}.$$

Embed the secret bits using Equation 4. The illustration is shown in Figure 2.

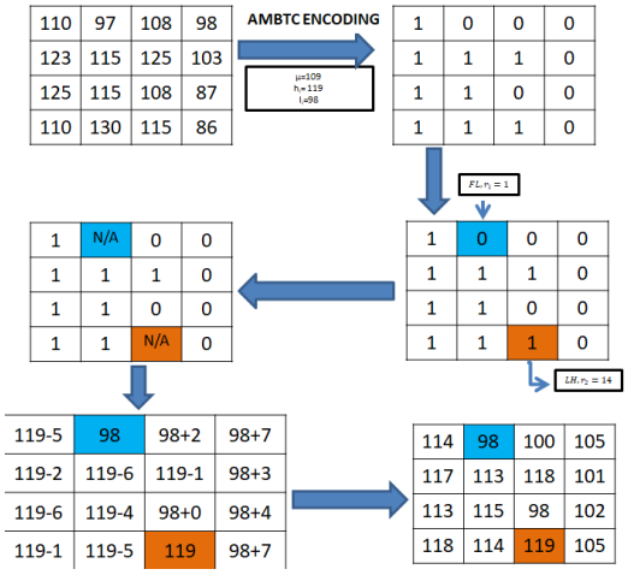


Figure 2: Illustration of SP1

Experimental Results.

Experimental results take six grayscale images size 512×512 (see Table 1). The images are divided into 4×4 non-overlapping blocks. The proposed method can embed 56 bits in each 4×4 pixel block, while the difference between the two quantization levels is more significant than 32.

2.2 Survey Paper 2 (SP2)

Lin *et al.* [13] divide the embedding procedure into three stages. The flowchart of the embedding procedure of Chen *et al.*'s method is shown in Figure 3. The detailed process of Lin *et al.*'s scheme is as follows:

Step 1. AMBTC Encoding

Firstly, manage an AMBTC encoding procedure to obtain a high mean table, low mean table, and bitmap table, H, L, and BM, respectively. Secondly, calculate the different values of each table with their neighbor with inverse "S" scan path as shown in Figure 4.

Step 2. Block Classification

Classify the current block into non embeddable or embeddable blocks by using condition. If the difference (D_i) is more significant than 2^m $(|D_i| > 2^m)$, it is classified as an embeddable block where m is the length of the binary representation of absolute D_i. However, when (D_i) is less than 2^m $(|D_i| < 2^m)$, the block is classified as non-embedding.

Step 3. Secret Embedding

The secret message is embedded in the mean for the embeddable block as a code stream following equation:

$$D_i' = \begin{cases} 1||Sign||m\ bits\ of\ |D_i|||b_1b_2b_3, & \text{if } |D_i \le 2^m \\ 0||V_{i_2}, & \text{if } |D_i > 2^m \end{cases}$$

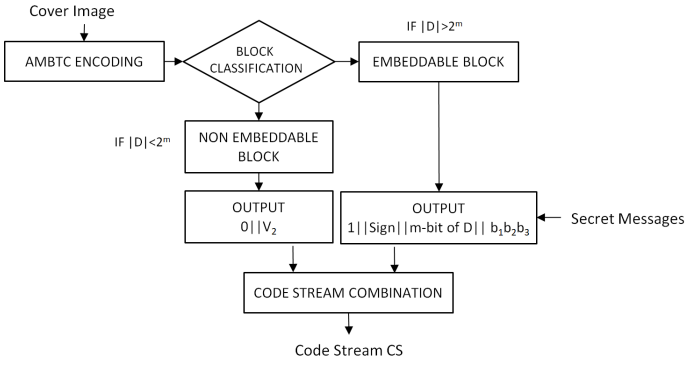


Figure 3: The Flowchart of Lin's method [13]

Experimental Results

Experimental results take six grayscale images size 512×512 (see Table 2). The images are divided into 4×4 non-overlapping blocks under m=5.

Table 1: Experimental results of SP1

Parameter	Lena	F16	Sailboat	Girl	Toys	Barbara	Average
PSNR	32.59	32.22	30.16	33.30	32.12	28.82	31.54
Hiding Capacity	324.548	267.386	432.796	388.780	292.250	449.876	359.272
BPP	1.23	1.01	1.65	1.48	1.11	1.71	1.36

Table 2: Experimental results of SP2

Parameter	Lena	Baboon	Peppers	F16	GoldHill	Boat	Average
PSNR	33.97	26	33.4	31.95	33.16	31.15	31.6
Hiding Capacity	87.199	86.902	88.771	88.648	89.569	88.321	88.235

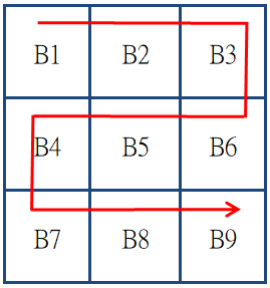


Figure 4: Inverse "S" scan path

2.3 Survey Paper 3 (SP3)

Horn *et al.* [6] method consist of three embedding phases. The secret bits are embedded in the high and low quantizers in the first phase using QVD and LSB methods. In the second phase, the secret bits are concealed by changing the order of a high and low quantizer. Lastly, in the third phase, the secret bits are hidden in the bitmap of the smooth block by direct substituting. The flowchart of the embedding procedure of Horn *et al.*'s technique is shown in Figure 5. The detailed process of Lin *et al.*'s scheme is as follows.

The specific procedure of [6] of the data hiding method is below.

1st Phase: QVD and LSB Substitution

1) AMBTC encoding to obtain (H, L, BM).
2) Shear the quantization levels (L, H) into (Q_L, Q_H) and (R_L, R_H) based on the equations.

$$\begin{aligned} Q_L &= L \div 4 \text{ and } Q_H = H \div 4 \\ R_L &= L \bmod 4 \text{ and } R_H = H \bmod 4. \end{aligned}$$

3) Determine the embedding capacity by comparing the absolute difference $|d|$ with Table 3. Suppose $|d|$ falls in the interval $r_j = r_j^L + r_j^H$, n_j data bits are collected and converted to decimal value b:

$$d = Q_H - Q_L$$

4) Calculate the new difference value d' and the new quotients pairs Q'_L, Q'_H by following the following equations:

$$\begin{aligned} d' &= r_j^L + b \\ (Q'_L, Q'_H) &= \begin{cases} (Q_L - \lfloor \frac{d'-d}{2} \rfloor, Q_H + \lceil \frac{d'-d}{2} \rceil), \\ \qquad \text{if } d \text{ is even} \\ (Q_L - \lceil \frac{d'-d}{2} \rceil, Q_H + \lfloor \frac{d'-d}{2} \rfloor), \\ \qquad \text{if } d \text{ is odd} \end{cases} \end{aligned}$$

5) Apply LSB substitution to (R_L, R_H) and obtain $(R'L, R'H)$.
6) Merge (Q'_L, Q'_H) and (R'_L, R'_H) back to (L', H'):

$$\begin{aligned} L' &= Q'_L \times 4 + R'_L, \\ H' &= Q'_H \times 4 + R'_H \end{aligned}$$

2nd Phase: Quantization Level Swapping

1) If $Q'_H > Q'_L$: when the secret data is '1' change the order of $(L', H', BM) \rightarrow (H', L', BM)$;
2) When the secret data is '0', keep (L', H', BM). Where BM is the bitmap value.
3) If $Q'_H = Q'_L$, cannot embed the secret data.

3rd Phase: Bitmap Replacement

1) Calculate the modified difference $|d_p| = |H' - L'|$:

   If $|d_p| < d_{th}$**:** Smooth block, replace BM with secret bits, where d_{th} is the threshold.

   If $|d_p| > d_{th}$**:** Complex block, the BM is non-embeddable.

2) Write the modified block to I' and repeat Steps 2-7 until all blocks are embedded.

Table 3: The capacity of SP3

Intervals	(0, 3)	(4, 7)	(8, 15)	(16, 31)	(32, 63)
Hiding Capacity	2	2	3	4	5

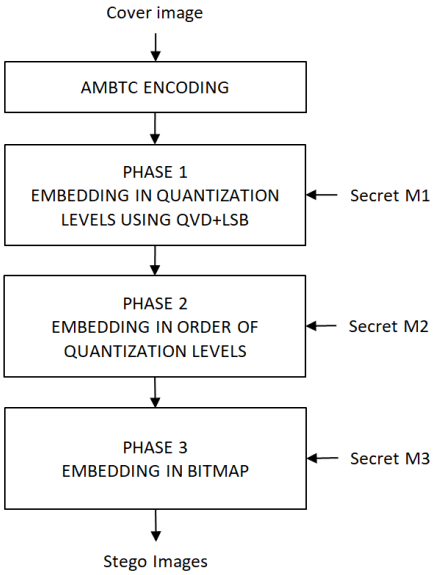


Figure 5: The flowchart of the Horng's method [6]

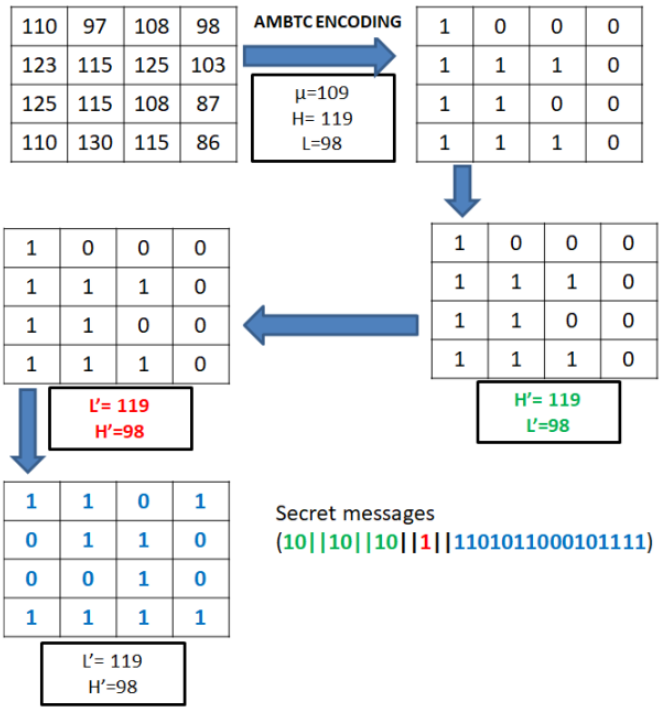


Figure 6: Illustration of SP3

2.3.1 Illustration of SP3

Suppose we have 4×4 of the size original image and 23 bits of secret messages. Secret messages (10||10||10||1||1101011000101111). The illustration of SP3 is shown in Figure 6.

1st Phase: QVD and LSB Substitution. Embed 6 bits:

1) AMBTC encoding to obtain (119, 98, 1000111011001110).

2) Shear the quantization levels (L, H) into (Q_L, Q_H) and (R_L, R_H) based on the equations.

$$\begin{aligned} Q_L &= 98 \; div \; 4 = 24 \;\&\; Q_H = 119 \; div \; 4 = 29 \\ R_L &= 98 \bmod 4 = 2 \;\&\; R_H = 119 \bmod 4 = 3. \end{aligned}$$

3) Determine the embedding capacity by comparing the absolute difference $|d|$ with Table 3.
Suppose $|d|$ falls in the interval $r_j = r_j^L + r_j^H$, n_j data bits are collected and converted to decimal value b:

$$d = Q_H - Q_L = 29 - 24 = 5.$$

4) Calculate the new difference value d' and the new quotients pairs Q'_L, Q'_H by following the following equations:

$$\begin{aligned} d' &= r_j^L + b = 4 + 2 = 6 \\ (Q'_L, Q'_H) &= (24 - \lfloor \frac{6-5}{2} \rfloor, 29 + \lceil \frac{6-5}{2} \rceil) \\ &= (24, 30). \end{aligned}$$

5) Apply LSB substitution to (R_L, R_H) and obtain (R'_L, R'_H):

$$\begin{aligned} R_L &= 2 = (10)_2 \rightarrow (10)_2 = 2 \\ R_H &= 3 = (11)_2 \rightarrow (10)_2 = 2. \end{aligned}$$

6) Merge (Q'_L, Q'_H) and (R'_L, R'_H) back to (L', H'):

$$\begin{aligned} L' &= 24 \times 4 + 2 = 98, \\ H' &= 30 \times 4 + 2 = 122. \end{aligned}$$

2nd Phase: Quantization Level Swapping. 0 or 1 bit embedded.
If $Q'_H > Q'_L$: When the secret data is '1' change the order of $(L', H', BM) \rightarrow (H', L', BM)$;

$$(H', L', BM) = (98, 122, 1000111011001110)$$

3rd Phase: Bitmap Replacement. 0 or 16 bits embedded.
Calculate the modified difference $|d_p| = |H' - L'| = |98 - 122| = 24$.

2.3.2 Experimental Results

Experimental results take six grayscale images size 512×512 (see Table 4). The images are divided into 4×4 non-overlapping blocks, under threshold=16.

2.4 Survey Paper 4 (SP4)

Lee *et al.*'s [10] method consists of three stages of embedding procedures. The first stage is data embedding at the quantization level. The second is data embedding in bitmaps. And the third stage is data embedding in order of quantization. The flowchart of Lee *et al.*'s method [10] is shown in Figure 7. The fully explained three stages are as follows.

1st stage: Data embedding at the quantization level.

1) AMBTC encoding to obtain (H, L, BM).
2) Generate a turtle-shell matrix based on [14] and map each block's quantization levels (H, L) into the matrix.
3) Map S1 (2 bits) to the position table yields the corresponding value, and the value is mapped to the position (H', L') closest to the quantification of the shell matrix. Modify (H, L) to (H', L') to match the secret data S1.

2nd stage: Data embedding in the bitmap.

1) Calculate the difference value $D = |L' - H'|$ to classify whether the block is smooth or complex.
2) When $D < TH$, the block is a smooth block. Therefore, the bitmap B_m of the block can hide the secret data. Substituting S2 (16 bits) for the original bitmap B_m forms a new bitmap B'_m. When the block is complex, not an embeddable block.

3rd stage: Data embedding in order of quantization level.

If $H' > L'$: When the secret data is '1' change the order of $(H,' L', BM) \rightarrow (L,' H', BM)$; When the secret data is '0' $(H,' L', BM)$. Where BM is the bitmap value.

If $H' = L'$: It cannot embed the secret data.

2.4.1 Illustration of SP4

Suppose we have 4×4 of the size original image and 21 bits of secret messages. Secret messages (1010||1011101011000101||1). Threshold=20. The illustration of SP4 is shown in Figure 8.

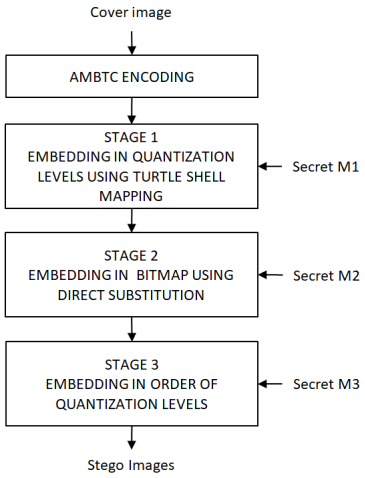


Figure 7: The flowchart of Lee et al's scheme [10]

1st stage: Data embedding at the quantization level.

1) AMBTC encoding to obtain (119, 98, 1000111011001110).
2) Generate a turtle-shell matrix based on [14] and map the quantization levels (119, 98) into the matrix, as shown in Figure 9(a).
3) Map S1=1010 (4 bits) to the position table yields the corresponding value, as shown in Figure 9(b). Modify (119, 98) to (117, 98) to match the secret data S1.

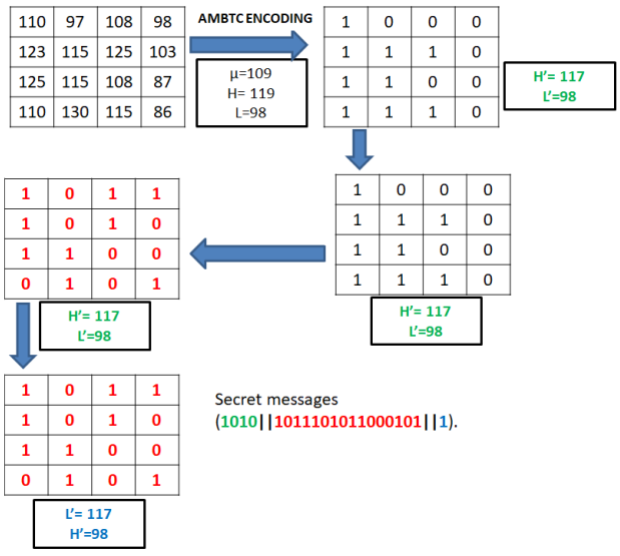


Figure 8: Illustration of SP4

2nd stage: Data embedding in the bitmap.

1) As the difference value $D = |L' - H'| = |98 - 117| = 19$, so that the block is smooth.

Table 4: Experimental results of SP3

Parameter	Airplane	Baboon	Boat	Lena	Pepper	Zelda	Average
PSNR	29.91	25.97	29.05	30.16	30.29	31.49	29.47
Hiding Capacity	313.825	210.959	297.787	315.188	326.352	336.462	299.595
BPP	1.19	0.80	1.12	1.20	1.21	1.28	1.14

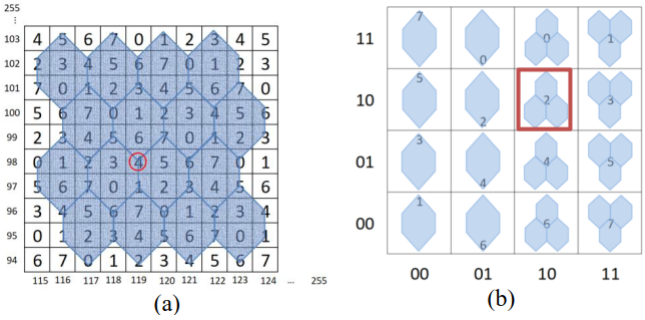


Figure 9: Turtle shell mapping

2) Substituting S2=1011101011000101 (16 bits) into the bitmap B_m.

3rd stage: Data embedding in order of quantization level.
As 117 > 98 and the secret data is '1' we change the order of (117, 98, 1011101011000101) → (98, 117, 1011101011000101).

2.4.2 Experimental Results

Experimental results take six grayscale images size 512 × 512 (see Table 5). The images are divided into 4 × 4 non-overlapping blocks. Threshold=20.

2.5 Survey Paper 5 (SP5)

Kumar *et al.* [9] classify the image block using two predefined thresholds, thr1 and thr2. The aim is to categorize the block into three types: Smooth, Less_Complex, and Highly_Complex. The detailed process of Kumar *et al.*'s scheme is as follows:

Step 1. AMBTC Encoding.
Firstly, manage an AMBTC encoding procedure to obtain high quantizer, low quantizer, and bitmap, H, L, and BM, respectively.

Step 2. Block Classification
Calculate the difference value $D = |L - H|$ to classify whether it is a Smooth, Less_Complex, and Highly_Complex block under the condition.

If $D < thr_1$, the block is Smooth.

If $thr_1 < D < thr_2$, the block is Less_Complex.

If $D > thr_2$, the block is Highly_Complex.

Step 3. Secret Embedding

- For a Smooth block, directly substitute the bitmap BM with the secret bits.
- Recalculate the high quantizer and low quantizer using the equation:

$$H' = \frac{1}{t-q}\sum_{p_i \in G_0} p_i$$

$$L' = \frac{1}{q}\sum_{p_i \in G_1} p_i$$

- For the Less_Complex block, embed the secret bits as follows:
  - Substitute the bitmap in an even location and calculate the hamming distance between the replaced bitmap and the original bitmap (h_1).
  - Substitute the bitmap in an odd location and calculate the hamming distance between the replaced bitmap and the original bitmap (h_2).
  - If $h_1 > h_2$, modify into H', L', and BM'.
  - If $h_1 < h_2$, flip all the BM value.
- For the Highly_Complex block,
  Embed the secret bits into the high and low quantizer.

Experimental Results
Experimental results take six grayscale images size 512 × 512 (see Table 6). The images are divided into 4 × 4 non-overlapping blocks. Threshold-1=10, and Threshold-2=25.

3 Comparisons

This section summarizes the performance of five survey papers (SP1-SP5) in terms of image quality and hiding capacity. First, we select five images used by the five survey papers, including two images used by the five survey papers, namely Lena and F16, and three images used by the four survey papers, namely Baboon, Pepper, and Boat.

Table 7 compares the image quality of SP1-SP5 in the PSNR parameter. Again, the image quality of SP1 outperforms that of others. At the same time, SP2 and SP4

Table 5: Experimental results of SP4

Parameter	Airplane	Baboon	Barbara	Boat	Lena	Pepper	Average
PSNR	31.01	26.67	28.52	29.92	30.76	31.06	29.65
Hiding Capacity	293.178	191.962	241.025	277.265	299.247	307.359	268.339
BPP	1.11	0.73	0.91	1.05	1.14	1.22	1.02

Table 6: Experimental results of SP5

Parameter	Lena	Baboon	Plane	Peppers	Boats	Barbara	Average
PSNR	31.43	27.23	30.37	31.36	29.20	31.19	30.79
Hiding Capacity	215485	140130	212684	221134	182575	202957	203198
BPP	0.82	0.53	0.81	0.84	0.70	0.77	0.78

image quality rank second and third, respectively, with PSNR values greater than 30, except for the Baboon image. Furthermore, the image quality in SP3 and SP5 is comparable.

Figure 10 depicts the image quality representation. Image Lena and Pepper outperform other images in all survey papers (SP1-SP5), with PSNR values greater than 30. The image F16 and Boat come next, with the Baboon image having the lowest image quality.

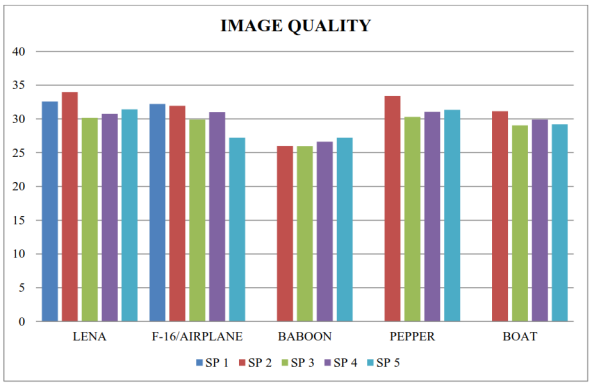


Figure 10: Image quality representation

Table 8 compares the hiding capacity of SP1-SP5 in the number of bits parameter. Again, the image quality of SP3 outperforms that of others. While SP1, SP4, and SP5 rank second, third, and fourth in hiding capacity, respectively. Furthermore, SP2 has the lowest hiding capacity.

The hiding capacity is depicted in Figure 11. Again, the image-hiding capacity of Lena and Pepper outperforms other images. The images F16 and Boat follow, with the Baboon image having the least amount of hiding capacity.

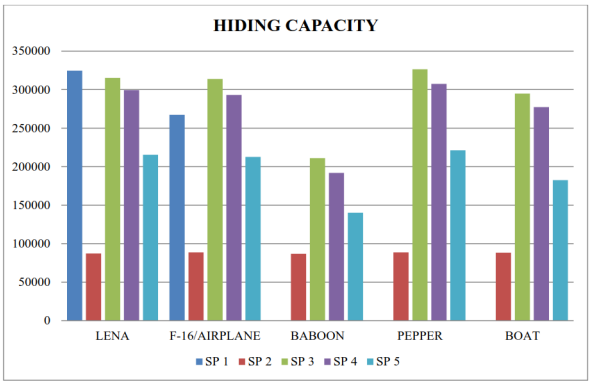


Figure 11: Hiding capacity representation

4 Future Research

In general, the five survey papers divide image blocks into two types: smooth and complex blocks. The smooth block is categorized as an embeddable block; it can embed more secret bits depending on the block size. However, complex blocks are categorized as non-embeddable; they cannot hide a secret. It could be a future work opportunity. How to hide secret bits in complex blocks for increasing hiding capacity without distorting the image. One concept that can be a solution is the combination theory as in the [21].

The embedding in the quantization level procedure is the second issue. The image in the AMBTC decoding procedure will be affected by changes in the high and low quantizer values. Therefore, we must tread carefully when implementing a method that will result in significant value changes. The Least Significant Bit is one of the concepts with the least amount of value change.

The same as the second issue, in the order of quantization level, when the secret bit is "1", the order of quantization is flipping. So it will affect the image distortion so that the quality of the image is poor. So that we have to focus more on this stage by managing new methods to

Table 7: Image quality comparison

Scheme	Lena	F16/Airplane	Baboon	Pepper	Boat
SP 1	32.59	32.22	0	0	0
SP 2	33.97	31.95	26	33.4	31.15
SP 3	30.16	29.91	25.97	30.29	29.05
SP 4	30.76	31.01	26.62	31.06	29.92
SP 5	31.43	27.23	27.23	31.36	29.2

Table 8: Hiding capacity comparison

Scheme	Lena	F16/Airplane	Baboon	Pepper	Boat
SP 1	324548	267386	0	0	0
SP 2	87199	88648	86902	88771	88321
SP 3	315188	313825	210959	326352	294787
SP 4	299247	293178	91962	307359	277265
SP 5	215485	212684	140130	221134	182575

handle this issue.

5 Conclusions

This paper thoroughly examines data-hiding research in the AMBTC compressed image domain. All survey papers (SP1-SP5) include secret messages in one of three AMBTC compressed image structures. It can be hidden in the quantization level, bitmap, and quantization level order. Least Significant Bit substitution, turtle shell matrix, hamming distance, and pixel value differencing are all possible methods. The comparison of experimental results shows that it is impossible to achieve both hiding capacity and image quality simultaneously. In the future, we hope to find a method that utilizes the AMBTC compressed image structure to embed more secret bits while maintaining image quality.

Acknowledgments

The Ministry of Science and Technology partially supported this research, Taiwan (ROC), under contract no.: MOST 109-2221-E-468-011-MY3, MOST 108-2410-H-468-023, and MOST 108-2622-8-468-001-TM1.

References

[1] C. C. Chen, C. C. Chang, C. C. Lin, and G. D. Su, "TSIA: A novel image authentication scheme for AMBTC-based compressed images using turtle shell based reference matrix," *IEEE Access*, vol. 7, pp. 133746-133761, 2019.

[2] Y. Chen, C. Chang, C. Lin, and Z. Wang, "An adaptive reversible data hiding scheme using AMBTC and quantization level difference," *Applied Sciences*, vol. 11, no. 635, 2021.

[3] K. Datta, B. Jana, and M. Dalui, "Two-layers robust data hiding scheme for highly compressed image exploiting AMBTC with difference expansion," *Journal of King Saud University - Computer and Information Sciences*, vol. 34, no. 8, pp. 5240–5260, 2022.

[4] O. Evsutin, A. Melman, R. Meshcheryakov, and S. Member, "Digital steganography and watermarking for digital images: A review of current research directions," *IEEE Access*, vol. 8, pp. 166589–166611, 2020.

[5] W. Hong, J. Wu, D. C. Lou, X. Zhou and J. Chen, "An AMBTC authentication scheme with recoverability using matrix encoding and side match," *IEEE Access*, vol. 9, 2021.

[6] J. Horng, "Steganography using quotient value differencing and LSB substitution for AMBTC compressed images," *IEEE Access*, vol. 8, pp. 129347–129358, 2020.

[7] Y. Hu, G. S. Member, and W. Yang, "Learning end-to-end lossy image compression: A benchmark," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 8, pp. 4194–4211, 2022.

[8] A. J. Hussain, A. Al-fayadh, and N. Radi, "Image compression techniques: A survey in lossless and lossy algorithms," *Neurocomputing*, vol. 300, pp. 44–69, 2018.

[9] R. Kumar, D. Kim, and K. Jung, "Enhanced AMBTC based data hiding method using hamming distance and pixel value differencing," *Journal of Information Security and Applications*, vol. 47, pp. 94–103, 2019.

[10] C. Lee, C. Chang, and G. Li, "A data hiding scheme based on turtle-shell for AMBTC compressed images," *KSII Transactions on Internet and Information Systems*, vol. 14, no. 6, pp. 2554–2575, 2020.
[11] M. D. Lema and O. R. Mitchell, "Absolute moment block truncation coding and its application to color images," *IEEE Transactions on Communications*, vol. 32, no. 10, pp. 1148–1157, 1984.
[12] C. Lin and Y. Huang, "Novel image authentication scheme for AMBTC-compressed images," in *International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, pp. 134-137, 2014.
[13] C. C. Lin, T. S. Nguyen, C. C. Chang, and W. C. Chang, "Efficient reversible data hiding scheme for AMBTC-compressed images," *Applied Sciences*, vol. 11, 2021.
[14] Y. Liu, C. Chang, and T. Nguyen, "High capacity turtle shell-based data hiding," *IET Image Processing* , vol. 10, no. 2, pp. 130–137, 2016.
[15] D. Ou, and W. Sun, "Reversible AMBTC-based secret sharing scheme with abilities of two decryptions," *Journal of Visual Communication and Image Representation*, vol. 25, no. 5, pp. 1222–1239, 2014.
[16] P. Shiu, W. Tai, J. Jan, C. Chang, and C. Lin, "An interpolative AMBTC-based high-payload RDH scheme for encrypted images," *Signal Processing: Image Communication*, vol. 74, no. December 2018, pp. 64–77, 2019.
[17] G. D. Su, C. C. Chang, and C. C. Lin, "A high capacity reversible data hiding in encrypted AMBTC-compressed images," *IEEE Access*, vol. 8, pp. 26984–27000, 2020.
[18] N. Subramanian and O. Elharrouss, "Image steganography: A review of the recent advances," *IEEE Access*, vol. 9, pp. 23409–23423, 2021.
[19] Y. Sun, C. Yang, X. Yan, Y. Lu, and L. Sun, "Robust secret image sharing scheme resistance to maliciously tampered shadows by AMBTC and quantization," *Gene Expression Patterns*, vol. 45, no. 110, p. 119267, 2022.
[20] H. Y. Wang, H. J. Lin, X. Y. Gao, W. H. Cheng, and Y. Y. Chen, "Reversible AMBTC-based data hiding with security improvement by chaotic encryption," *IEEE Access*, vol. 7, pp. 38337–38347, 2019.
[21] N. Wu and M. Hwang, "Development of a data hiding scheme based on combination theory for lowering the visual noise in binary images," *Displays*, vol. 49, pp. 116–123, 2017.
[22] X. Wu and C. Yang, "Invertible secret image sharing with steganography and authentication for AMBTC compressed images," *Signal Processing: Image Communication*, vol. 78, pp. 437–447, 2019.
[23] X. Wu and C. Yang, "Partial reversible AMBTC-based secret image sharing with steganography," *Digital Signal Processing*, vol. 93, pp. 22–33, 2019.
[24] C. N. Yang, X. Wu, M. J. Chung, and X. Zhang, "AMBTC-based secret image sharing by simple modular arithmetic," *Journal of Visual Communication and Image Representation*, vol. 84, p. 103482, 2022.
[25] W. Zheng, "A novel adjustable RDH method for AMBTC-compressed codes using one-to-many map," *IEEE Access*, vol. 8, pp. 13105–13118, 2020.

Biography

Kurnia Anggriani received BS degree in Informatics from University of Bengkulu, Indonesia in 2011, and the MS degree in Informatics from Bandung Institute of Technology, Indonesia in 2014. Currently she is taking Ph.D degree in Asia University, Taiwan. Her current research interests include steganography and image processing.

Nan-I Wu received a Ph.D. degree in the Institute of Computer Science and Engineering from Nation Chung Hsing University (NCHU), Taichung, Taiwan, in 2009. From 2010 to 2011, he was a post-doctoral research fellow at the Academia Sinica Institute of information science. He was an assistant professor at the Department of Animation and Game Design, TOKO University (Taiwan), during 2011-2018 and an associate professor during 2018-2019. Now he is an associate professor at the Department of Digital Multimedia, Lee-Ming Institute of Technology (Taiwan) since 2019 and also the Director of the eSports Training Centre since 2020. His current research interests include game design, eSports training/magagement, multimedia processing, multimedia security, data hiding, and privacy-preserving. He published more than 10 international journal papers (SCI) and conference papers.

Min-Shiang Hwang received M.S. in industrial engineering from National Tsing Hua University, Taiwan in 1988, and a Ph.D. degree in computer and information science from National Chiao Tung University, Taiwan, in 1995. He was a distinguished professor and Chairman of the Department of Management Information Systems, NCHU, during 2003-2011. He obtained 1997, 1998, 1999, 2000, and 2001 Excellent Research Awards from the National Science Council (Taiwan). Dr. Hwang was a dean of the College of Computer Science, Asia University (AU), Taichung, Taiwan. He is currently a chair professor with the Department of Computer Science and Information Engineering, AU. His current research interests include information security, electronic commerce, database, data security, cryptography, image compression, and mobile computing. Dr. Hwang has published over 300+ articles on the above research fields in international journals.

A Multistage Dynamic Defense Method for Evolutionary Games

Zhiyong Luo, Yutong Cao, Weiwei Song, and Jie Li
(Corresponding author: Luo Zhiyong)
School of Computer Science and Technology, Harbin University of Science and Technology
Harbin 150080,China
Email: luozhiyongemail@sina.com
(Received Jan. 27, 2022; Revised and Accepted Oct. 13, 2022; First Online Oct. 22, 2022)

Abstract

This paper proposes a multistage dynamic defense method for evolutionary games to address the challenge of accurately sensing unknown and homeopathic attacks on each node in the network and effectively accomplishing dynamic security. The method combines the replicator dynamic equations and the characteristics of the attack-defense game adversarial process to establish a discrete multistage offense-defense game model, and then quantifies the gains and equilibrium solutions for the model, simulates the multistage offense-defense game process under the high selection rate of the attack and defense strategy, and calculates the maximum objective function values of both sides. By analyzing these function values, the security situational awareness of the whole network nodes is completed to predict future security situations and system maintenance. Experimental comparisons show that the model approach has high operational efficiency and better defensive performance to ensure system integrity and other advantages.

Keywords: Dynamic Interaction; Multistage; Replicator Dynamic Equations; Situational Awareness

1 Introduction

With the widespread use of the Internet, people are becoming more and more inseparable from computers in their production and life. While we are currently improving network connectivity through constantly updated network technologies, we are also witnessing an era of unprecedented cyber attacks. Ensuring the confidentiality, integrity and availability of data, devices, networks and users has become critical. Most cybersecurity research has focused either on targeting specific vulnerabilities or proposing specific defense algorithms to defend against well-defined attack scenarios. Most of the defense techniques, which are static and passive, however, cannot effectively accomplish dynamic security in the face of unknown attacks, transient attacks in the network. Although such network security research is important, little attention has been paid to the dynamic interaction between attackers and defenders.

Game theory has the characteristics of goal opposition, relationship non-cooperation, and strategy dependence, which are consistent with the basic characteristics of network attack and defense [15]. Therefore, people apply game models to the field of network security to reason about intrusion intentions, targets and strategies. Traditional games are built on the premise of complete rationality of decision makers, which does not match with the actual attack and defense and reduces the effectiveness of models and methods. Considering the limited rationality of the attack and defense sides in the real network, evolutionary game theory is applied to the study of the attack and defense process [11].

Yao *et al.* proposed a multi-variant execution architecture-based CFI (MVX-CFI). MVX-CFI is an execution architecture-based, dynamic, and transparent CFI (control integrity) implementation that effectively captures the direction of control flow throughout the software runtime and detects illegal path shifts caused by malicious behaviors such as attacks [20]. Tian *et al.* modeled sequential attacks in complex networks as a partially observable Markov decision process (POMDP). Then a POMDP reinforcement learning method is proposed to analyze the dynamic robustness of complex networks under sequential attacks when the network information is incomplete [19]. Foschini *et al.* analyzed and identified the correct detection and mitigation strategies for DoS attacks in IT/OT networks. Provided DoS detection and mitigation strategies in business-centric IT/environment and production-centric operational technology (IT/OT) networks [6]. Hu *et al.* extend the game model to a novel two-way signaling game model and proposed an algorithm to identify the refined Bayesian equilibrium. Based on the calculated payoffs, the optimal strategy choices of the attacking and defending parties during the game are analyzed [9]. Hsieh *et al.* characterized the equilibrium of the underlying game and used the Bayesian dual Metropolis-

Hastings algorithm to estimate the model. And further extended the model to incorporate unobserved heterogeneity and showed that ignoring unobserved heterogeneity leads to biased estimation in simulation experimentsHsieh2020A. Zhang *et al.* proposed heterogeneity to improve the security of web servers with mimetic constructs and pointed out the importance of quantifying heterogeneity. A quantification method applicable to quantify heterogeneity is proposed, by which the factors affecting the heterogeneity of a mimetically constructed Web server are analyzed. A new method is provided for the quantitative assessment of mimetic defense [21]. Chen *et al.* innovatively used users as third-party participants in the moving target defense game and combined Stackelberg game and Markov model to construct a non-reciprocal three-party game to determine the optimal strategy for moving target defense. The proposed model can balance the cost and benefit of defenders and users, avoid excessive defense and inappropriate defense, and effectively achieve intelligent defense strategy decision [4]. Huanruo *et al.* conducted a comprehensive survey of the current state-of-the-art quantitative evaluation. MTD techniques based on the software stack model for classification. Then, a specific review and comparison of existing quantitative evaluations of MTD is presented [10]. Sengupta *et al.* provided a comprehensive survey of MTD and implementation strategies from the perspective of complete network system architecture. Discussed how various MTDs are implemented, analyzed MTD testbeds and case studies, and classified MTDs according to qualitative and quantitative metrics of security and performance effectivenessSengupta2020A. Zhang *et al.* proposed a user-oriented anti-censorship approach that significantly increases the cost to attackers. Representing Web services as mobile nodes forms a moving target defense strategy by using mobile IPv6 [22]. Sharma *et al.* proposed the random host and service multiplexing technique, RHSM. This technique uses shuffling of IP addresses and port numbers and aims to obfuscate the true identity of hosts and services at the network and transport layers to defend against network reconnaissance and scanning attacks [17].

The above studies have established different network security risk assessment models based on game theory, but they are too ideal for the establishment of attack and defense models, which cannot truly reflect the possibility of attackers' choice of target networks and attack methods, and do not quantify the probability situation of strategy selection for the intentions of both attackers and defenders. In this paper, a multistage dynamic attack and defense model based on evolutionary game theory is proposed under the premise of information asymmetry between offensive and defensive, and the main work and innovations are as follows.

1) Combining evolutionary game theory and Nash equilibrium, each round of attack and defense will adjust the strategy according to the newly acquired information and vested interests, and the game process goes through many iterations to finally reach the dynamic equilibrium state;

2) Based on the limited amount of information, in response to the attacker's attack intention, the defender releases false defense signals to cope with the complex network with changing security elements and improve the predictive capability of the defense situation;

3) Considering the complex factors affecting the attacker's attack behavior, the attack probability is calculated from three indicators: signal deception cost, attack cost and defense cost, which more realistically reflects the attacker's situation in the actual network.

2 Construction of Multistage Dynamic Game Model

2.1 Multistage Dynamic Game Process Analysis

In the traditional network attack and defense process, attackers mainly uses network attacks or detection methods to obtain information about the target network, so as to achieve the analysis and penetration of the vulnerability of the target system and finally find the most appropriate network attack strategy to make the optimal network attack benefit [7]. Due to the natural asymmetry of the network attack process, the attacker is able to actively obtain the information about the target network and carry out the network attack at any time, while the defender is often in a passive defense state [5]. In order to change the passive defense situation, the defender takes the initiative to release defense signals so that the network attacker cannot judge the authenticity of the information, thus influencing the attacker's choice of attack strategy and making the network defense passive to active. The attack and defensive game process analysis of defense signals is a key factor in the attacker's analysis to determine the type of defender and the choice of action decisions [13].

In the initial stage of the game, the defender deceives and restricts the attacker by releasing a false defense signal so that the attacker cannot obtain the real state of the target system; the attacker forms an initial a priori probability judgment of the defender by combining the detection behavior and intelligence collection of the target in the early stage, and then forms a posteriori probability of the defense type based on the defense signal released by the defender, inspired by Bayes' law. Define the replicator dynamic equation to select the optimal network attack strategy, thus completing the initial process of the game. After the initial stage of the game, the defender releases the deception signal suitable for this stage again and selects the corresponding optimal defense strategy; the attacker takes the posterior probability of the defense

type obtained in the previous stage as the prior probability of this stage, and combines the defense signal received in this stage to derive the posterior probability of the defender type in this stage and selects the corresponding optimal attack strategy. The specific process is shown in Figure 1.

Analyzed from the perspective of the discount factor, the defense signal is strongest in the first phase with a discount factor expressed as $\delta_T = 1$. With the progress of the attack and defense process, the attacker absorbs the learning experience so that the effectiveness of the defense deception signal decreases, with$0 < \delta_i < 1$.

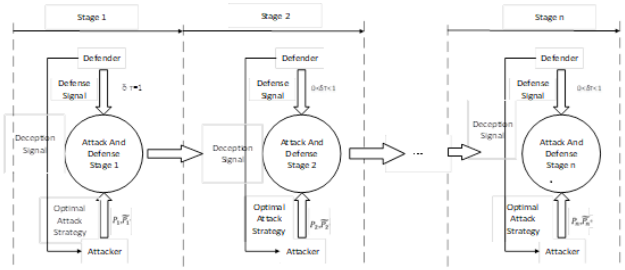


Figure 1: Multi-stage network attack and defense game model

2.2 Construction of Multistage Dynamic Game Model

Definition 1. *Multistage Dynamic Attack And Defense Game Model MDADGM as an octet(N,T,S,θ,M,δ, P,U). The details are as follows:*

1) *$N = (N_A, N_D)$ denotes the set of game participants, where N_A is the defender and N_D is the attacker;*

2) *T denotes the number of stages in the multi-stage game process, i.e., $T = 1..., n$;*

3) *$S = (S_A, S_D)$ denotes the set of game strategies, where S_D represents the set of strategies of the defender, $S_D = \{S_{D_j} | j = 1, 2..., n\}$;$S_A$ represents the set of strategies of the attacker, $S_A = \{S_{A_i} | i = 1, 2..., m\}$;*

4) *θ denotes the set of game participant types, and $\theta_D = \{\theta_{D_i} | i = 1, 2..., n\}$ indicates the set of defender types, and the defense type is the defender's private information, There is only one type of attacker, namely $\theta_A = (\eta)$ [12];*

5) *M denotes the defense signal space, satisfying $M \neq \oslash$,$M = \{m_k | k = 1, 2, ...\}$. Defenders for the purpose of deterring, deceiving and luring attackers, the network defense signal and the defender's true type may not match;*

6) *δ is the discount factor, which indicates the degree of discount of the defense signal in the game stage T compared with the previous game stage, and satisfies $0 < \delta < 1$;*

7) *P represents the set of game beliefs, where $P_A = \{P_A(\theta_{D_1}), P_A(\theta_{D_2}) ... P_A(\theta_{D_n})\}$ denotes the attacker's prior probability for different defense types. $\widetilde{P_A} = \widetilde{P_A}(\theta_{D_i} | m_k)$ denotes the posterior probability calculated by the attacker combining the defense signal and the prior probability [2];*

8) *U denotes the set of gain functions of both attackers and defenders, where, U_A denotes the attacker's gain function and U_A denotes the defender's gain function.*

2.3 Quantification of Game Gains

The game theory is applied to the analysis of network attack and defense, and the quantification of game gain is the key to determine the accuracy of the final game result.

Definition 2. *Attack cost (AC) represents the economic, time, hardware and software, and human resources spent by the attacker due to the choice of attack strategy.*

Definition 3. *Defense cost (DC) indicates resources such as economy, time, hardware and software equipment, and labor, as well as the impact of degradation of service quality due to the defender's choice of a defense strategy.*

Definition 4. *System damage cost (SDC) represents the damage caused to the system after an attacker initiates an attack.*

Definition 5. *Defense effectiveness ε indicates the effectiveness of defense policy d against attack a. When the attack can be completely blocked, $\varepsilon(a, b) = 1$; when the defense strategy is ineffective, $\varepsilon(a_i, b_j) = 0$; in other cases, $0 < \varepsilon(a_i, b_j) < 1$ [?].*

Definition 6. *Signal deception explore (SDE) represents the cost of the defender to deceive the attacker by releasing false signals. If the signal matches the true type of the defender, the SDE is zero. The SDE is relatively quantified according to the gap between the true and false defense information and is expressed as an integer value within the interval $[0, 100]$ [18]. In general, the greater the gap between defender and defensive signal, the more difficult and costly to camouflage. The classification and quantification of SDE are shown in Table 1.*

During network attack and defense, the attacker aims to minimize the cost of attack while maximizing the cost of system loss, while the defender minimizes the cost of defense, the cost of network deception, and the cost of system loss. Using different strategies for offensive and defensive confrontation generates different offensive and defensive payoffs [3]. At each stage, the reward expectations for attackers and defenders are as shown in Equations (1) and (2).

$$U_A = (1 - \varepsilon)SDE + DE - AC \tag{1}$$

Table 1: Quantitative table of signal spoofing costs

Defender Real Type	Defender Spoofing Signals	SDE Level	Quantitative Allocation
High level defender	High level defender	SDE0	10
	Medium level defender	SDE1	40
	Low level defender	SDE2	70
Medium level defender	High level defender	SDE1	10
	Medium level defender	SDE0	40
	Low level defender	SDE2	70
Low level defender	High level defender	SDE2	70
	Medium level defender	SDE1	40
	Low level defender	SDE0	10

$$U_D = AC - (1-\varepsilon)SDE - DC \tag{2}$$

3 Equilibrium Solution of the Game Process and Optimal Strategy Selection

3.1 Nash Equilibrium

Definition 7. *(Nash Equilibrium) Given a network attack and defense game model MDADGM (N,T,S,θ,M,δ, P,U), S_{A_i} is the attacker strategy and S_{D_j} is the defense system strategy. The strategy (S_A^*, S_D^*) is a Nash equilibrium [14] when and only when the strategy is optimal for both the attacker and the defender, that is, it satisfies:*

$$\forall i, U_A(S_A^*, S_D^*) \geq U_A(S_{A_i}^*, S_D^*) \tag{3}$$

$$\forall j, U_D(S_A^*, S_D^*) \geq U_D(S_A^*, S_{D_j}^*) \tag{4}$$

Theorem 1. *(Nash Equilibrium Existence) Given a network attack and defense game model MDADGM (N,T,S,θ,M,δ, P,U), there exists at least one Nash equilibrium.*

The network attack and defense game model MDADGM (N,T,S,θ,M,δ, P,U) is a matrix-type game whose set of attack and defense strategies S, and gain functions are finite, so the network attack and defense game model MDADGM is a finite game. Nash proves that every finite game has a Nash equilibrium using the immobility theorem, so the network attack and defense game model MDADGM has a stable Nash equilibrium, that is, given a network attack and defense game model, it must be possible to solve its optimal dynamic equilibrium attack and defense strategy.

3.2 Replicator Dynamic Equation

In the set of game participants, when the attacker receives the defense deception signal $m_k \in M$ the posterior probability $P(\theta_{D_i}|m_k)$ is calculated in conjunction with the prior probability, and at the same time, the defender is able to anticipate that the attacker will pick the inferentially dependent optimal attack strategy $S_A^*(m_k)$ based on the network deception signal m_k released by itself, so the defender picks the optimal network defense policy $S_D^*(m_k)$ that maximizes the expected gain of defense. At this point, we can interpret that the probability of the chosen strategy changes for both sides through learning from the previous round of the game.

Theorem 2. *(Replicator Dynamics Equation) The growth rate of the number of individuals choosing strategy S will be less than 0 if the gain obtained by individuals choosing strategy S is less than the average gain of the population and vice versa. The replicator dynamic equation is a dynamic differential equation that reflects the frequency when a strategy is adopted in the set [23]. It is usually expressed by equation 5:*

$$\frac{\mathrm{d}x_i}{\mathrm{d}i} = x_i(U_{S_{A_i}} - \overline{U_A}) \tag{5}$$

where x_i indicates the proportion of strategy S adopted in the set,$U_{S_{A_i}}$ indicates the return when strategy S is adopted, and $\overline{U_A}$ represents the average return.

3.3 Optimal Strategy Selection

In choosing the optimal defense strategy, the equilibrium state of the multistage game is analyzed and the evolutionary equilibrium is solved by using the replicator dynamic equation. The equilibrium solution steps of the evolutionary game can be obtained as follows:

1) Based on the defense signal released by the defender, the probability of the strategy selected by the attacker on the strategy set is p, and the attacker's replicator dynamic equation is:

$$A(P) = \frac{\mathrm{d}p_i(t)}{\mathrm{d}x} = p(U_{S_{A_i}} - \overline{U_A}) \tag{6}$$

Among them, $U_{S_{A_i}} = \sum_{j=1}^{n} q_j a_{ij}$, $\overline{U_A} = \sum_{i=1}^{m} p_i U_{S_{A_i}}$, $U_{S_{A_i}}$ represents the gain function when the attacker chooses the attack strategy a_{ij}.

2) The probability that a defender chooses a strategy on the strategy set is q. The defender's replicator dynamics equation is:

$$D(q) = \frac{\mathrm{d}q_j(t)}{\mathrm{d}t} = q(U_{S_{D_j}} - \overline{U_D}) \tag{7}$$

3) Quantification of Utility Functions
In a multistage game, as the attacker gradually determines the type of the defender, the gain obtained gradually decreases based on the initial gain. Therefore, in this paper, a discount factor δ_T is introduced to calculate the future gain based on the original gain function U. The calculation is as follows:

$$\begin{cases} U_D^k = U_D^k + \sum_{h-1}^{k-1} \delta_T U_D^h \\ U_A^k = U_A^k + \sum_{h-1}^{k-1} \delta_T U_A^h \end{cases} \tag{8}$$

k=$\{1, 2...T\}$.

4) Equilibrium Solution
According to the evolutionary game equilibrium state, the replicator dynamic equations of the attacker and defender should be equal to 0. The game equilibrium solution should satisfy the following equation:

$$\gamma = \begin{bmatrix} maxU_A^k \\ maxU_D^k \\ A(p) = 0 \\ D(p) = 0 \end{bmatrix} \tag{9}$$

k=$\{1, 2...T\}$.

By solving the above equations together, the set of strategy choices under the evolutionary equilibrium state $(S_{D_j}^k, S_{A_i}^k)$ can be obtained. According to the evolutionary game theory, the offensive and defensive strategies at this time are the best choices for both attackers and defenders.

3.4 The Equilibrium Solution Process of Multistage Attack and Defense Game

Suppose the defender type θ_D is divided into high level defender θ_h, medium level defender θ_m, and low level defender θ_l, the corresponding defense signal space M has m_h, m_m,m_l,the defense strategy space is $m^*(\theta)$, the attacker type θ_A =(η), the attack strategy space is $\{S_{A_1}, S_{A_2}, S_{A_3}\}$, the defense type prior probability is P_A, and the attack and defense gains are(U_A, U_D).

1) When T=1, enter the first stage of the attack and defense game
For the initial stage of the attack and defense game, nature selects the defender type with probabilities $P_A(m_h)$, $P_A(m_h)$, $P_A(m_h)$. If a network defender releases a false defense signal m_1, when the attacker receives the signal m_1, it will probabilistically correct the defender type with the posterior probability $\{\widetilde{P_A}(\theta_{D_1}|m_1), \widetilde{P_A}(\theta_{D_2}|m_1), \widetilde{P_A}(\theta_{D_n}|m_1)\}$ to discriminate the defender type as $\{\theta_{D_1}, \theta_{D_2}, \cdots, \theta_{D_n}\}$. Similarly, when the attacker receives the defense signal m_i, the defender type is determined with the posterior probability of $\{\widetilde{P_A}(\theta_{D_1}|m_i), \widetilde{P_A}(\theta_{D_2}|m_i), \widetilde{P_A}(\theta_{D_n}|m_i)\}$. The attack and defense signal game tree is shown in Figure 2.

According to the Nash equilibrium existence theorem, given the participant types and limited attack and defense strategies, both attackers and defenders are maximizing the expected gain as much as possible, and combining Theorem 2, we obtain the replicator dynamic equations $A(p)$ and $D(p)$ for both attackers and defenders. The set of equations is constructed by associating the gain functions to obtain the optimal set of attack and defense strategies $(S_A^*(m), S_D^*(m))$ at this stage. In the initial stage of the game, the attacker cannot analyze the actual type of the defender from the pre-confrontation process, and there is no signal attenuation effect of the false defense signal released by the defender. At this time, $\delta_1 = 1$.

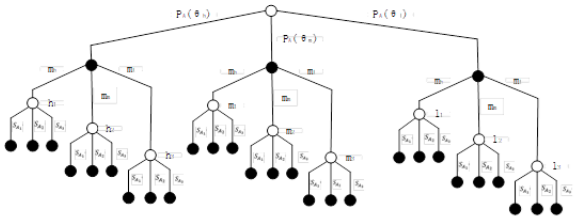


Figure 2: Attack and defense signal game tree

2) When T = 2, enter the second stage of the attack and defense game
By analyzing and comparing the game process and outcome information of the previous stage, the attacker enhances the analysis and screening ability of the defense signals, so from the second stage, $0 < \delta_2 < 1$,the signal discounting effect starts to appear. The attacker takes the posterior probability obtained in the previous stage as the prior probability in this stage, and then combines the false defense signal released in this stage to obtain $\delta_2 \widetilde{P_A}(\theta_{D_2}|m_2))$. The set of most attack and defense strategies for the second stage is again obtained by Nash equilibrium and replicator dynamic equations.

3) When T = n, enter the n stage of the attack and defense game
When the number of game phases T tends to be large or even infinite and the defender releases false signals

more often, $\delta^{T-r-1} \approx 0$, where r represents the number of phases in which the defender releases real defense signals. According to the basic theory of signal game, the game stage becomes a static game with incomplete information, as shown in the game tree in Figure 3. The method of solving the incomplete information static game can be found in the literature [1].

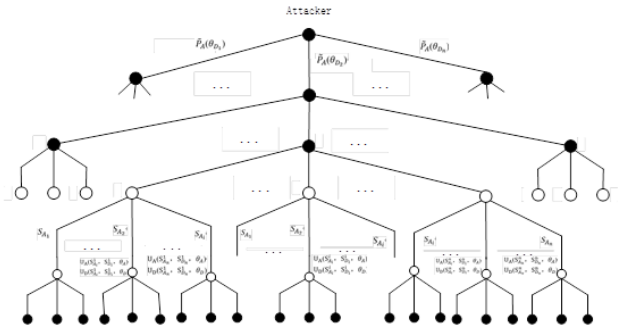


Figure 3: Incomplete information static attack and defense game tree

3.5 Algorithm Design

Based on the analysis process of the above multistage network attack and defense game, the optimal defense strategy selection algorithm is designed as follows.

Algorithm Optimal defense strategy selection algorithm for multi-stage network attack and defense games.

Input (N,T,S,θ,M,δ, P,U)
Output Optimal defense strategy $S_{D_j}^K$

1) Initialize MDADGM=(N,T,S,θ,M,δ, P,U), S_{A_i};

2) Build a defensive operational space S_D= $\{S_{D_j}^k | 1 \leq k \leq T, 1 \leq j \leq n\}$, attack Space S_A= $\{S_{A_i}^k | 1 \leq k \leq T, 1 \leq i \leq m\}$ and k denotes the number of game stages;

3) Build a defender type space $\theta_D = (\theta_h, \theta_m, \theta_l)$ and attacker type space$\theta_A = (\eta)$;

4) Initialize defender defense signal space $M \neq \oslash$, $M = (m_h, m_m, m_l)$;

5) while $(\exists \varepsilon \&\& S_{D_j} \in S_D \&\& S_{A_i} \in S_A)$ { //Calculate earnings

$$U_D^k = U_D^k + \sum_{h-1}^{k-1} \delta_T U_D^h$$
$$U_A^k = U_A^k + \sum_{h-1}^{k-1} \delta_T U_A^h$$

for $(i = 1; i \leq m; i++)$// Learn the set of offensive and defensive strategies to construct replicator dynamic equations

$$A(p) = \frac{\mathrm{d}p_i(t)}{\mathrm{d}t} = p(U_{S_{A_i}} - \overline{U_A});$$
$$D(q) = \frac{\mathrm{d}q_j(t)}{\mathrm{d}t} = q(U_{S_{D_j}} - \overline{U_D}).$$

6) for$(k = 1; k \leq T; k++)$// Building a network attack and defense game at different stages
{if$(\delta_T > 0)$ // Discount factor to solve the replicator dynamic equation solution
{ $U_D^k = U_D^k + \sum_{h=1}^{k-1} \delta_T U_D^h$
$U_A^k = U_A^k + \sum_{h=1}^{k-1} \delta_T U_A^h$;}
while$(\exists maxU_A^k \&\& maxU_D^k)$
{ $A(p) = 0$;
$D(p) = 0$;}
if$(\delta_T = 0)$
{// Incomplete information game solve
}}

7) end for;

8) return $\{S_{D_j}^1, S_{D_j}^2, S_{D_j}^3 ... S_{D_j}^n\}$.

4 Experiment and Analysis

4.1 Description of Experimental Environment

In order to further illustrate the effectiveness of the proposed active defense model and its related algorithms, a simulation experiment is carried out by deploying the experimental scenario shown in Figure 4. The experimental environment mainly consists of network defense devices, network servers, file servers, database servers, client servers, etc., and mainly Windows and Linux operating systems are installed. The security defense rules are to restrict the access requests from hosts outside the system (including attackers), and stipulates that they can only access the network server; the file server and the network server are allowed to access the database server. However, with the help of a multi-step attack process, the attacker is able to gain access to the application server and the database server.

4.2 Calculation of Game Profit

The method for analyzing routing files, vulnerability databases and defense strategies in literature [7] is combined with the information on atomic attacks given in literature [16], as shown in Table 2.

In this network, the SQL database server exists with important data, and a_3 can be considered as the attacker's intrusion intent to set the attack strategy using the scanned vulnerability information, the relationship between vulnerabilities, host and server information, network configuration and other data. To simplify the calculation, we only consider high-level defenders and low-level defenders. The descriptions of the attack and defense

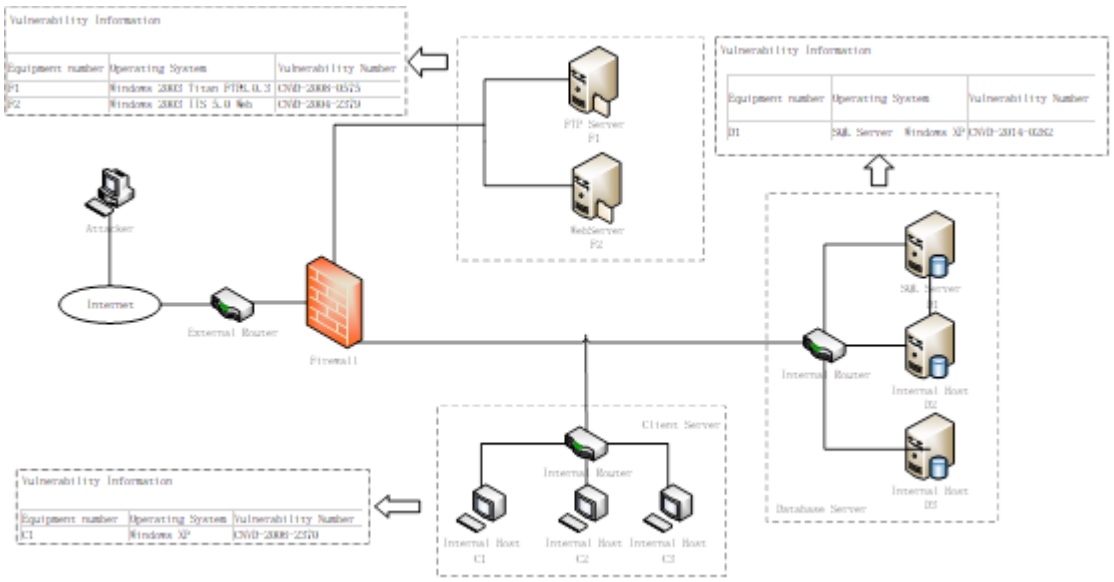


Figure 4: Experimental environment network topology

Table 2: Atomic attack strategy

Serial Number	Description of the Atomic Attack Strategy	Permission	AL
a1	Attack SSH on FTP sever	root	10
a2	Unsigned firmware update	root	10
a3	Database rights	root	11
a4	LPC to LSASS process	root	11
a5	Remote code execution	uesr	8
a6	Attack address blacklist	user	9
a7	Oracle TNS listener	user	7
a8	Remote buffer overflow	user	8
a9	Install SQL listener program	access	6

atomic strategies are given by using the Nessus scanning experimental information system with reference to the attack and defense classification method of MIT Lincoln Laboratory [8] and the National Information Security Vulnerability Database are shown in Table 3.

The real defender level is set to low defender and the defensive signal is set to high defender, that is, SDE = 70. The set of attack strategies in this system includes a1 and a2, and the set of defensive strategies includes d1 and d2. According to the calculation formula given in Section 2.3, the quantization of the attack and defensive strategies in the experiment are:

$$\begin{aligned}(a_{11}, d_{11}) &= (45, -45)\\ (a_{12}, d_{12}) &= (68, -68)\\ (a_{21}, d_{21}) &= (22, -22)\\ (a_{22}, d_{22}) &= (45, -45).\end{aligned}$$

When the defender uses strategies d1 and d2, respectively, the expected gain are:

$$\begin{aligned}U_{DS_1} &= pd_{11} + (1-p)d_{21} = -45p - 22(1-p)\\ U_{DS_2} &= pd_{12} + (1-p)d_{22} = -68p - 45(1-p).\end{aligned}$$

The average gain for the defenders is:

$$\begin{aligned}\overline{U_D} &= qU_{DS_1} + (1-q)U_{DS_2}\\ &= q[45p - 22(1-p)] + (1-q)[-68p - 45(1-p)].\end{aligned}$$

For the defensive strategy DS_1, the probability that the defender chooses this strategy is a function of time and its dynamic rate of change can be expressed as:

$$\begin{aligned}D(q) &= \frac{\mathrm{d}q(t)}{\mathrm{d}t}\\ &= q[-45p - 22(1-p) - 45pq + 22q(1-p)\\ &\quad +68p(1-q) + 45(1-q)(1-p)].\end{aligned}$$

Similarly, the expected gains obtained by the attacker using strategies a1 and a2 are:

$$U_{AS_1} = qa_{11} + (1-q)a_{12} = 45q + 68(1-q) \qquad (10)$$

Table 3: Atomic defense strategy

Description of the Atomic Attack Strategy	θ_{D_H}			θ_{D_L}		
	S_{D_1}	S_{D_2}	S_{D_3}	S_{D_4}	S_{D_5}	S_{D_6}
Limit packets from ports	√	×	√	×	√	√
Install Oracle patche	×	√	√	√	×	×
Reinstall listener program	√	√	×	×	×	√
Uninstall delete Trojan	√	√	√	√	√	×
Renew data(root)	√	×	×	×	×	√
Restart database sever	×	√	√	√	×	×
Limit SYN/ICMP packets	√	√	√	×	√	√
Add physical resourse	√	×	√	√	√	×
Repair database	√	√	×	×	×	×

$$U_{AS_2} = qa_{21} + (1-q)a_{22} = 22q + 45(1-q) \quad (11)$$

The average gain for attackers is:

$$\begin{aligned}\overline{U_A} &= pU_{AS_1} + (1-p)U_{AS_2}p[45q + 68(1-q)] \\ &\quad +(1-p)[22q + 45(1-q)].\end{aligned}$$

The dynamic rate of change of the selection strategy AS_1 is:

$$\begin{aligned}A(p) &= \frac{\mathrm{d}p(t)}{\mathrm{d}t} = p(U_{AS_1} - \overline{U_A}) \\ &= p[45q + 68(1-q) - 45pq - 68p(1-q) \\ &\quad -22q(1-p) + 45(1-q)(1-p)].\end{aligned}$$

Since the optimal defense strategy is generated in equilibrium, this experiment analyzes and solves the equilibrium of the evolutionary game in the final stage. At this point, the effect of the false defense signal released by the defender on the game outcome completely disappears, $\delta_T = 0$, and the objective function is equal to the gain function, that is, $\gamma = [A(p), D(q)] = 0$. Then, the data are re-substituted into the algorithm for validation, and the evolutionary stable strategy is obtained from the image.

4.3 Example Analysis

The attacker's attack and control of the network is reflected in the control of each node component of the network. This example expresses the attacker's invasion status of the network as the attacker's access authority to each node component of the network, and the level of authority is divided into: no access permissions, remote access permissions, local user access permissions, and root access permissions. At the beginning of the game phase, because in the actual network application to define the specific start and end nodes is a rather difficult thing, this paper starts the attacker to launch an attack on different nodes to invade the test, and the defender releases the deception factor after discovering it. Taking the atomic attack strategy a_3 as an example, the attack paths are $S_{A_1} = \{a_1, a_2, a_5\}$, $S_{A_2} = \{a_3, a_5, a_7\}$, $S_{A_3} = \{a_4, a_6, a_9\}$. Assuming that a total of three stages of the games are required, the first and second stages are both offensive and defensive signal games, where the role of false defense signals decreases continuously and the probability of the attacker inferring that the defender is a low defense level increases continuously, and in the third stage, the attacker is able to completely screen false signals, the role of defensive signals disappears, and the attack and defensive games degenerate into incomplete information static games. The game tree is formed as shown in Figure 5.

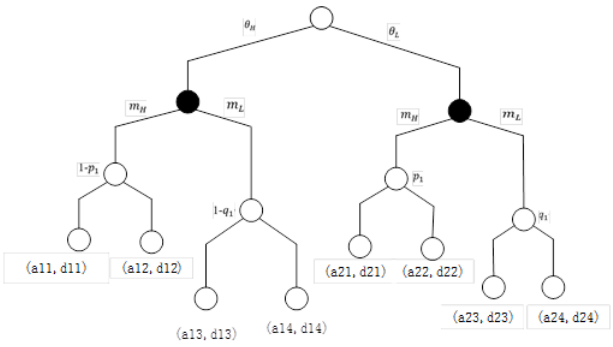


Figure 5: Game tree

Figure 6 shows the analysis of the evolutionary stability strategy of the defender with different probabilities of the attacker's choice of strategy. According to the defender's replicator dynamic equation, the defender's evolutionary stability strategy choice has the following cases.

When p=0, there exists $D(p) = 0$ for any defensive strategy selection probability. when $p \neq 0$, $D(q)$ changes significantly, as shown in Figures 6 and 7. When the slope of the tangent line of the curve is positive, the evolutionary stable strategy of the defender is obtained. Therefore, in Figure 6, when p¿0, $d_1(q = 1)$ is the defender's evolutionary stable strategy. In Figure 7, when p¡0, $d_2(q = 0)$ is the defender evolutionary stable strategy. And since p cannot be less than 0, d_2 is the best defense strategy

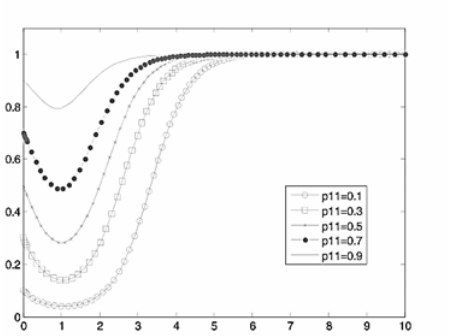


Figure 6: Evolutionary stability strategy selection diagram

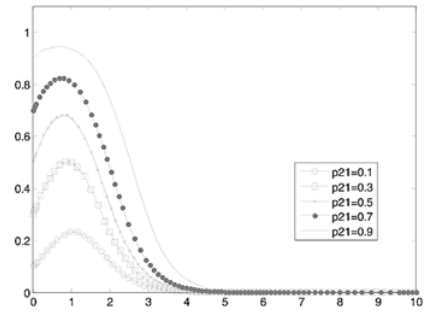


Figure 7: Evolutionary stability strategy selection diagram

at this point. To demonstrate that the analytical results are consistent with the results in the real scenario, we use Matlab to perform simulations to obtain the evolutionary game equilibrium. The results are shown in Figure 8. The horizontal axis represents time and the vertical axis represents the initial value of q. From the figure, we can see that the system finally reaches stability at q=1 regardless of the initial probability p. It is proved that d_1 is the optimal defense strategy solution and the proposed MDADGM model is feasible and effective.

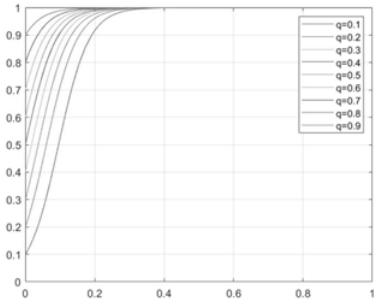


Figure 8: Curve simulation diagram

4.4 Comparison of Methods

The reachability probability of each attribute node in the attack and defense process is the main indicator for network security risk assessment, and the prediction of attack paths can provide network administrators with a basis for intrusion defense. In order to verify the superiority of the model in this paper, in the same network environment, it is necessary to conduct simulation research on different test models. The specific operation process is as follows: Five models were selected for testing: model A in this paper, model B in literature 5, model C in literature 7, model D in literature 16, and model E in literature 23.

Figure 9 shows the running time of the five algorithms in a single stage, three stages, six stages, and 12 stages, respectively, under the same network environment. Models B and C also use the signaling game model to describe the causal relationship between network attack behaviors. However, because their evaluation indicators of vulnerabilities are too single and do not take into account the costs and benefits of attacks, the vulnerability utilization of both does not truly reflect the exploited situation of vulnerabilities in the network. Model D and model E are solved by refined Bayesian equilibrium, but they also lack multiple metrics to evaluate the calculation. As can be seen from Figure 9, the accuracy of the evaluation models in this paper is significantly better than the other four models, because this paper calculates the probability of atomic attacks from multiple indicators and evaluates them more accurately.

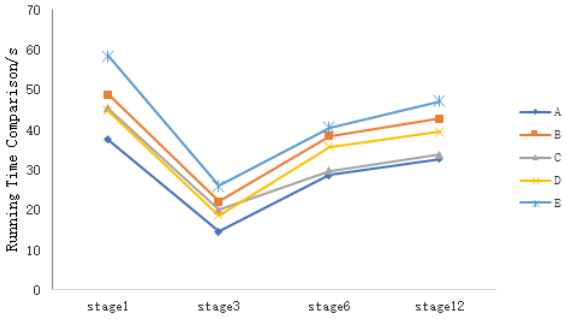


Figure 9: Running time comparison

The total plenum value E(Q) of the minimum critical strategy set of different defense methods is used as an important index to evaluate the defense performance of different methods, where the larger the value of E(Q) is, the better the defense performance of the algorithm is. The specific experimental comparison results are given below.

As can be seen in Figure 10, the total weight of the minimum critical policy set of model A in this paper is higher than that of model B in reference 5, model C in reference 7, model D in reference 16, and model E in reference 23. The E(Q) values of model B in reference 5 and model C in reference 7 are about 65% and 55%, while the E(Q) value of the method in this paper is as high as

about 80%. The attack and defense revenue strategy is fully considered in the atomic attack probability, and a discount factor is introduced to balance the error caused by deception signals in the multistage attack and defense process. Experimental research shows that the defense method proposed in this paper has better defense performance.

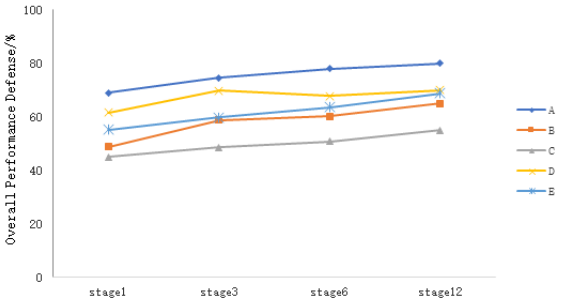


Figure 10: Comparison of overall defense performance results

5 Conclusion

For the traditional intrusion detection, firewall and other passive defense technologies can not cope with the increasingly prominent network security problems. This paper applies noncooperative signal game theory to network attack and defense analysis, makes full use of defense signals to confuse and deter attackers, constructs a multistage network deception game model, and conducts in-depth research on the signal deception mechanism existing in the process of network attack and defense. The research results can provide an effective modeling method for network security situation prediction and theoretical guidance for the application of defense deception in the field of network security. In response to the strong learning ability of attackers and the endless attacks, the next step needs to further consider how to adjust the defense strategy to make the network deception signal achieve better deception effect. And consider more influencing factors to optimize the gain function to make it more suitable for the actual situation.

Acknowledgments

This study was supported by the Natural Science Foundation of Heilongjiang Province:LH2021F030. The authors gratefully acknowledge the anonymous reviewers for their valuable comments.

References

[1] M. Amin, S. Shetty, L. Njilla, D. K. Tosh, C. Kamhoua, "Hidden markov model and cyber deception for the prevention of adversarial lateral movement," *IEEE Access*, vol. PP, no. 99, pp. 1–1, 2021.

[2] J. U. Arshed, M. Ahmed, "Race: Resource aware cost-efficient scheduler for cloud fog environment," *IEEE Access*, vol. PP, no. 99, pp. 1–1, 2021.

[3] X. Chen, X. Liu, L. Zhang, C. Tang, "Optimal defense strategy selection for spear-phishing attack based on a multistage signaling game," *IEEE Access*, pp. 1–1, 2019.

[4] Z. H. Chen, G. Cheng, "Moving target defense technology based on stackelberg-markov non-peer-to-peer three-way game model," *Computer Science*, vol. 43, no. 3, p. 14, 2020.

[5] M. A. El-Zawawy, E. Losiouk, M. Conti, "Vulnerabilities in android webview objects: Still not the end!," *Computers & Security*, vol. 109, p. 102395, 2021.

[6] L. Foschini, V. Mignardi, R. Montanari, D. Scotece, "An SDN-enabled architecture for it/ot converged networks: A. proposal and qualitative analysis under ddos attacks," *Future Internet*, vol. 13, 2021.

[7] X. Hai, Z. Wang, Q. Feng, Y. Ren, H. Duan, "Mobile robot adrc with an automatic parameter tuning mechanism via modified pigeon-inspired optimization," *IEEE/ASME Transactions on Mechatronics*, vol. PP, no. 99, pp. 1–1, 2019.

[8] C. S. Hsieh, M. D. Knig, X. Liu, "A structural model for the coevolution of networks and behavior," *Review of Economics and Statistics*, pp. 1–14, 2020.

[9] Y. Hu, H. Zhang, Y. Guo, T. Li, J. Ma, "A novel attack-and-defense signaling game for optimal deceptive defense strategy choice," *Wireless Communications and Mobile Computing*, vol. 2020, no. 2, pp. 1–10, 2020.

[10] L. I. Huanruo, Y. Guo, S. Huo, G. Cheng, W. Liu, "Survey on quantitative evaluations of moving target defense," *Chinese Journal of Network and Information Security*, 2018.

[11] J. M. Huang, H. W. Zhang, "Optimal defense strategy selection based on improved replication dynamic evolutionary game model," *Communications Journal*, vol. 39, no. 1, pp. 170–182, 2018.

[12] M. F. Hyder, M. A. Ismail, "Securing control and data planes from reconnaissance attacks using distributed shadow controllers, reactive and proactive approaches," *IEEE Access*, vol. PP, no. 99, pp. 1–1, 2021.

[13] B. Liu and H. Wu, "Optimal d-facts placement in moving target defense against false data injection attacksn," *IEEE Transactions on Smart Grid*, vol. PP, no. 99, pp. 1–1, 2020.

[14] J. W. Liu, J. J. Liu, Y. L. Lu, B. Yang, K. L. Zhu, "Optimal defense strategy selection method based on network attack and defense game model," *Computer Science*, vol. 45, no. 6, pp. 117–123, 2018.

[15] Z. Y. Luo, X. Yang, J. H. Liu, R. Xu, "Network intrusion intention analysis model based on bayesian attack graph," *communications journal*, vol. 41, no. 9, pp. 160–169, 2020.

[16] D. Rotondo, H. S. Sánchez, V. Puig, T. Escobet, J. Quevedo, "A virtual actuator approach for the secure control of networked LPV systems under pulse-width modulated DoS attacks," *Neurocomputing*, vol. 365, no. 99, pp. 21–30, 2019.

[17] D. P. Sharma, J. H. Cho, T. J. Moore, F. F. Nelson, H. Lim, S. K. Dong, "Random host and service multiplexing for moving target defense in software-defined networks," in *ICC 2019 - 2019 IEEE International Conference on Communications (ICC)*, 2019.

[18] J. L. Tan, H. W. Zhang, H. Q. Zhang, H. Jin, C. Lei, "Optimal strategy selection method for moving target defense based on markov time game," *Journal of Communications*, vol. 41, no. 1, pp. 42–52, 2020.

[19] M. Tian, Z. Dong, X. Wang, "Reinforcement learning approach for robustness analysis of complex networks with incomplete information," *Chaos Solitons & Fractals*, vol. 144, no. 4-5, p. 110643, 2021.

[20] D. Yao, Z. Zhang, G. F. Zhang, J. X. Wu, "Mvx-cfi: a practical active defense architecture for software security," *Journal of Information Security*, vol. 5, no. 4, p. 11, 2020.

[21] J. X. Zhang, J. M. Pang, Z. Zhang, "Quantification method for heterogeneous web servers with mimicry construction," *software journal*, no. 2, p. 14, 2020.

[22] S. Zhang, "A moving target defense anti-censorship method based on mipv6," *Computer Applications and Software*, vol. 48, no. 6, pp. 2874–2883, 2019.

[23] Z. Zhang, R. Deng, P. Cheng, D. Yau, "Zero-parameter-information data integrity attacks and countermeasures in iot-based smart grid," *IEEE Internet of Things Journal*, vol. PP, no. 99, pp. 2327–4662, 2021.

Biography

Luo Zhiyong biography.Luo Zhiyong, male, was born in Shandong, China in July 1978.He is a professor at the School of Computer Science and Technology, Harbin University of Science and Technology. Research direction: computer network and information security, network optimization. words.

Cao Yutong biography. Cao Yutong, female, was born in Shandong, China in October 1997. She is a postgraduate student in the School of Computer Science and Technology, Harbin University of Science and Technology. Research direction: computer network and information security, network optimization, offensive and defensive games. optimization.

Song Weiwei biography.Song Weiwei, male, was born in May 1998 in Shanxi, China. He is a postgraduate student in the School of Computer Science and Technology, Harbin University of Science and Technology, Research direction: network security.

Li Jie biography.Li Jie is a lecturer in the School of Computer Science and Technology, Harbin University of Science and Technology. Research direction: network security.

Intrusion Detection Algorithm Based on Residual Neural Network

Zengyu Cai[1], Jingchao Wang[2], Jianwei Zhang[2], and Yajie Si[1]
(Corresponding author: Jianwei Zhang)

College of Computer and Communication Engineering, Zhengzhou University of Light Industry[1]
College of Software Engineering, Zhengzhou University of Light Industry[2]
Zhengzhou 450000, China
Email: mailzjw@163.com

(Received May 27, 2022; Revised and Accepted Oct. 17, 2022; First Online Oct. 22, 2022)

Abstract

The existing intrusion detection technology is not ideal for detecting new intrusions. To address the problem, this paper proposes a novel intrusion detection model based on residual neural networks, using the residual structure to fully mine the interdependence between network traffic features and predict the current network state. Compared with traditional neural networks, this approach can avoid model overfitting and has higher accuracy. Finally, this paper employs the NSL-KDD benchmark dataset to test the approach provided in this work and analyzes the effect of hyper-parameter on the model's performance. Experiments reveal that the method presented in this paper outperforms traditional neural networks.

Keywords: Intrusion Detection; Machine Learning; Residual Neural Network

1 Introduction

With the development of the Internet, people's life has become more and more convenient. However, while the Internet provides convenience to people's lives, it also poses a risk. With the constant updating and iteration of network technology, new cyberattacks continue to emerge. In the early years, most people used firewalls to defend against cyber attacks. But firewalls require human-made rules and can be easily bypassed. Therefore, intrusion detection techniques have been proposed to detect the cyberattacks missed by firewalls [1]. Although intrusion detection technology has achieved good results, traditional intrusion detection technology can no longer match current network protection needs due to the rapid iteration of cyberattack technology [13].

Intrusion detection technology is divided into two categories according to the data source: host-based intrusion detection technology and network-based intrusion detection technology. The former usually collects information from program execution records on the host system. It detect intrusions by monitoring host system calls, file modifications, and other activities. Although this intrusion detection method has a certain effect, it can only detect the host where the system is installed, and cannot observe other computer in the same network. Compared with host-based intrusion detection technollogy, the deployment of network-based intrusion detection technology is relatively simple. It can monitor intrusion behaviors across the entire network with a single deployment, saving time and the cost of deploying the system on each computer. It detects intrusions in the network by analyzing network traffic in real-time [12]. However, To identify whether an incursion occurs, traditional intrusion detection systems require a lot of regular or aberrant behavioral features. The size of the rule base has a substantial impact on its detection capability. The rule base takes a lot of work to maintain, and the detection impact of new and undiscovered cyberattacks is not ideal [18].

With the development of machine learning technology, more and more researchers introduce it to Intrusion detection [3]. Even though intrusion detection approaches based on machine learning are presently playing a role in many security situations due to the increasing rise of Internet data, traditional machine learning algorithms cannot meet today's security needs due to their limited learning capabilities. Different from traditional machine learning algorithms, deep learning methods mine the underlying rules of sample data and use multi-layer neural networks to adapt to higher-dimensional learning and prediction needs [2,20]. Now, it is being used to handle many complicated pattern recognition problem such as image classification, object detection, and machine translation. However, deep neural networks face the risk of high model complexity and model overfitting. This paper proposes a residual neural network-based intrusion detection model to handle this risk. Compared to the common neural network model, the intrusion detection model based on a residual neural network improved robustness and reduced overfitting effectively. Testing on the NSL-KDD testing

set suggests that the model we proposed can effectively increase the model's accuracy.

2 Related Work

2.1 Machine Learning-based Intrusion Detection Technology

Due to the advancement of machine learning, a growing number of people are using it in a variety of fields. Machine learning also plays a vital part in intrusion detection. The authors of [24] use a loosely assumed Hidden Naive Bayes method for multi-classification of traffic data. which uses a variety of methods to discretize the con-tinuous features and significantly improves the identification accuracy of abnormal traffic. The authors of [17] use continuous time Bayesian networks to model host-based intrusion detection and network-based intrusion detection, respectively. For the former, the method builds a hierarchical model and employs Rao-Blackwellized particle filtering to learn parameters. For the latter, the method develops a novel learning method to process system log files. In addition, The authors of [16] employ the principal component analysis approach to extract features, combine fuzzy technology to examine the degree to which the sample belongs to the assault sample, and then split the attack category using the K-Nearest Neighbor method. However, this method is less effective in the case of a relatively large amount of data. In addition to using classification methods to identify attack features, many people use regression methods. The authors of [6] used the simulation method to fit thousands of polynomial logistic regression models, found 13 risk factors that were significantly correlated, and used these risk factors to construct the final regression model. The authors of [4] developed a detection model based on outliers by using statistical methods combined with the regression method of binary classification and judged whether there was an attack by checking the fixed fields in the data packet. Although these methods per-form well in some scenarios, they are slightly weak for the current big data environ-ment.

2.2 Intrusion Detection Based on Deep Learning

Traditional machine learning methods are shallow learning methods. However, as network data grows in volume and complexity, the complexity and diversity of network data grow as well [1,4,14]. With the introduction of deep learning theory, many researchers use deep learning methods to replace traditional machine learning. Compared with classical machine learning methods, deep learning methods can mine the underlying hierarchical representation of data by building a multi-layer network structure and using high-dimensional complex connections. The authors of [15] use an unsupervised greedy learning algorithm to train and fine-tune deep belief networks to learn internal relational representations of high-dimensional input data. The authors of [23] developed an intrusion detection framework based on multi-scale convolutional neural networks and LSTM to learn features from multiple di-mensions of network traffic data, achieving outstanding detection results. The authors of [8] proposed a graph-based traffic classification model. This method constructs traffic data into a graph structure, takes IP addresses as vertices in the graph, and then establishes the relationship between each vertex, to analyze the attributes of the graph.

2.3 Residual Neural Network

The artificial neural network(ANN) was proposed as early as 1943 [22]. Over the decades, various versions have been developed, such as Convolutional Neural Net-works, Radial Basis Function Networks, and Recurrent Neural Networks. They're employed in a variety of applications, including data mining, image processing, and natural language processing. It imitates the learning ability of biological neural net-works and emphasizes the characteristics of neural networks in the human brain. In 2012, the success of AlexNet [11] in an image recognition competition led researchers to believe that the deeper the network, the higher the accuracy. But, with the improvement of computer computing power, deeper and deeper networks have been designed by people. He Kaiming et al found in the experiment that the accuracy of the network will not increase with the increase of depth, but will reach a maximum value, and then will greatly decrease as the network's depth increases, which is different from the previous conflicting views [19]. He Kaiming's team called this phenomenon "Degradation", and proposed a neural network Resnet with "Shortcut Connection", which won the championship in the 2015 image recognition competition.

3 Residual Neural Network-based Intrusion Detection Model

In this paper, the residual learning method is used to detect intrusion, and an intrusion detection model based on residual neural network is proposed. Specifically, the intrusion detection process is regarded as a classification problem, and the traffic features in the network are classified to judge whether there is an attack in the network.

3.1 NSL-KDD Benchmark Dataset

The KDD-CUP99 dataset is the basis for the NSL-KDD dataset. KDD-CUP99 is a 9-week collection of network connectivity statistics from a simulated US Air Force network. It is a classic intrusion detection data set, but there are problems such as data redundancy. The NSL-KDD is extracted from KDD-CUP99, and its training set and test set do not contain redundant records, so the classification model can be better trained and the classification

model can be prevented from being biased toward frequent records.

3.2 Data Preprocessing

The NSL-KDD dataset has some character features, but intrusion detection systems based on residual neural networks can only recognize numeric features. Therefore, choosing an appropriate method to quantify the character features in the dataset is critical to the performance of the experimental model. In addition, considering the huge difference between different features of the samples, the model may be affected by some large-valued features that are not important for key information mining, thus slowing down the convergence speed of the model. So, after quantifying, the data must be normalized to compress all feature values into the range 0-1. This facili-tates fast and stable convergence of the model.

Each sample in the NSL-KDD dataset has 41 feature values, with three categories of values being character types: 'protocol type', 'service', and 'flag'. There are three protocol kinds in 'protocol type', 70 service types in 'service', and 11 network con-nection types in "flag." This paper uses one-hot encoding to numerically process the three feature values indicated above, avoiding the impact of the size connection between the integers on the model training. For example, there are three values of protocol, namely 'tcp', 'udp' and 'icmp', and the corresponding numerical values are [0,0,1], [0,1,0] and [1,0, 0]. After numerical processing, the length of the eigenval-ue has changed from 41 to 122.

After quantifying, to avoid the value of some feature values being too large and having a bad influence on model training and testing, normalizing the feature values is required. The max-min normalization method is used to tackle the difficulties in this study. It has the following formula:

$$x_i' = \frac{x_i - min(x_0, x_1, ...x_n)}{max(x_0, x_1, ...x_n) - min(x_0, x_1, ...x_n)} \tag{1}$$

Among them, x_i represents a feature value, $min()$ represents the minimum value among the dataset, and $max()$ denotes the maximum value among the data set. x_i' represents the eigenvalue after normalization.

3.3 Construction of Intrusion Detection Model Based on Residual Neural Network

Inspired by Resnet, in this paper, an intrusion detection model based on residual neu-ral network is proposed. The model's input is a 122-dimensional vector, and its output is a 2-dimensional vector. The overall of model is illustrated in Figure 1, which is made up of 5 fully connected layers. This model uses the input of the previous layer and the output of the previous layer together as the input of the current layer in the third and fourth layers, rather than merely using the output of the previous layer as the input of the current layer.

The difficulty to execute identity transformation in deep neural networks is the cause of model degradation in typical deep neural networks. More and more network layers and activation functions cause the data to be mapped to a more discrete space, and it is difficult to make the post-data return to its original state. The emergence of the Shortcut Connection is a good balance between linear transformation and nonlinear transformation.

This paper does not directly use the form of Shortcut in [7] to directly add the two vectors together but concatenate the two by concatenation. The latter is more flexible than the former and can degenerate into the former under certain conditions. The structures without shortcut connection and those with shortcut connection added can be expressed as:

$$y_i = x_i w_i + b_i \tag{2}$$

$$y_i = concat(x_i, x_{i-1}) w_i + b_i \tag{3}$$

where i represents the number of the neural network layers, w_i is the neural network weight of this layer, and b_i is the bias. $concat()$ represents the concatenation operation. In Equation (2), the output of the network layer can only get information from x_i. But in Equation (3), the network layer's output can obtain both the previous layer's output and the previous layer?s input. The latter can reduce the negative effects of information loss during the $i-1$ layer computation. Expanding on Equation (3) as follows:

$$y_i = [x_i, x_{i-1}][w_\alpha, w_\beta]^T + [b_\alpha, b_\beta] \tag{4}$$

Among them, when $w_\alpha = w_\beta$ and $b_\alpha = b_\beta$, using the concatenate to realize the combination of two vectors has the same effect as using the direct addition method.

3.4 Model Training and Prediction

There are two steps in the model training process, the first is forward propagation and the second is backpropagation. In this paper, a single-layer network is given as an example to explain the process of backpropagation and forwarding propagation. The process of forwarding propagation can be defined as:

$$\hat{y} = \theta(W \cdot X + b) \tag{5}$$

The $\theta()$ represents the activation function, and the commonly used activation functions in deep learning are sigmoid, relu, and tanh. $\hat{y}$ representing the predicted value belonging to the sample X represents the set of feature values, W denotes the weight and b represents the bias. The loss computation must be performed when this forward propagation is done, and the loss function can be defined as:

$$Loss(X) = \sum_{(X,y)\in D} (y - \theta(W \cdot X)) \tag{6}$$

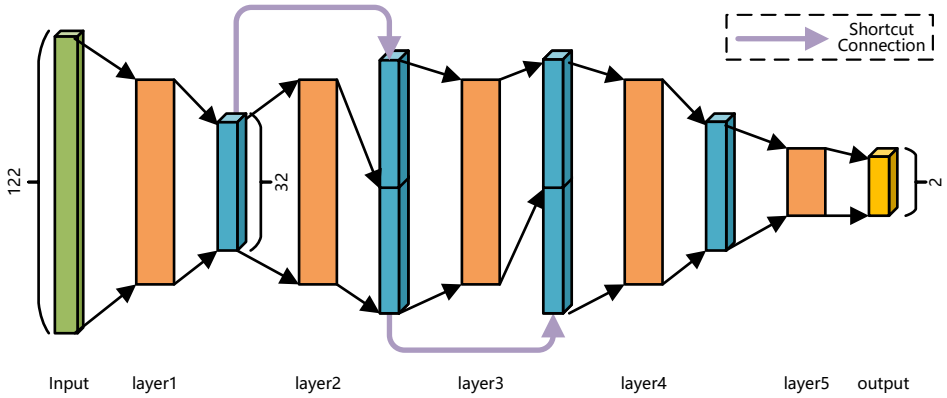


Figure 1: Intrusion Detection Model Based on Residual Neural Network

Where D represents the sample set. After getting the loss value, the process of backpropagation begins. First, the gradient is calculated, and the weights of the model are updated with a certain learning rate α. Its process can be defined as:

$$W \Leftarrow W - \alpha \frac{\partial Loss(X)}{\partial W} \tag{7}$$

In the model prediction process, only the process of forwarding propagation is included, and loss calculation and weight update are not required.

4 Experiment

To test the proposed residual neural network-based intrusion detection model in this research, we conduct simulation experiments using PyTorch 1.11 based on python3.8. All experiments were carried out on a PC with 8 TITAN RTX systems.

4.1 Experimental Dataset

The NSL-KDD dataset contains a total of 125,937 pieces of data, the size of it is just right for training the model. It will not cause the model to be underfitting due to too little, nor will it be necessary to formulate a sample screening strategy because of too much. Therefore, training the model with this dataset can compare well with the results of other models. This paper uses 10-fold cross-validation on the model in the experiment. The dataset is separated into ten sections in this work, nine of which are utilized to train the model and one of which is utilized for testing.

4.2 Hyperparameter Optimization

In this paper, to obtain the best performance, we use the grid-search for hyperparam-eter optimization. The method can evaluate every possible permutation of selected hyperparameters. This paper focuses on the selection of activation functions, opti-mizers, and batch sizes.

The activation function is an important part of ANN design, it directly affects the performance of a neural network. Each activation function has different effects on the overall performance and convergence of the neural network, so the choice of activation function is very important. In this paper, we choose the three most commonly used activation functions for experiments, namely Sigmoid, Rectified Linear Unit (ReLU), and Hyperbolic Tangent (tanh).

One cycle of learning and adjusting the network weights is called an Epoch, and the number of samples used in another iteration becomes the batch size. Different batch sizes affect the convergence speed and convergence effect of this model. In this paper, we choose 10, and 100 as the batch size for hyperparameter search. During the training process, the optimizer's choice has an impact on the best solu-tion for the model parameters.

A suitable optimizer can cause the model to fall into overfitting and achieve the global optimum. In this paper, we choose three optimizers such as Adam, SGD, and RMSprop for experiments.

Table 1 shows the performance of each combination of hyperparameters.

By adjusting the hyperparameters that act as points, this paper obtains the model with the highest accuracy of 98.27%. The activation function to achieve the optimal result is ReLU, the optimization method is Adam, and the batch size is 10.

4.3 Analysis of Model Training Process

During the model training process, we selected the precision, accuracy, and f1-score evaluation indicators. The accuracy and f1-score of the model are constantly rising

Table 1: The effect of different hyperparameters on model accuracy

Accuracy(%)	Precision(%)	Recall(%)	f1-score(%)	Activation	Optimizer	Batch Size
98.27	97.29	99.55	98.39	ReLU	Adam	10
97.73	96.50	99.34	97.90	ReLU	Adam	100
53.71	53.71	100	69.88	ReLU	SGD	10
53.28	53.28	100	69.52	ReLU	SGD	100
98.04	97.00	99.45	98.21	ReLU	RMSprop	10
98.03	96.95	99.45	98.18	ReLU	RMSprop	100
97.84	96.79	99.27	98.01	Sigmoid	Adam	10
97.57	96.69	98.88	97.75	Sigmoid	Adam	100
53.93	53.93	100	70.07	Sigmoid	SGD	10
53.76	53.76	100	69.92	Sigmoid	SGD	100
97.61	96.29	99.32	97.78	Sigmoid	RMSprop	10
97.48	96.58	98.80	97.68	Sigmoid	RMSprop	100
97.46	96.76	98.56	97.65	Tanh	Adam	10
97.72	96.80	99.00	97.89	Tanh	Adam	100
90.58	85.41	99.37	91.86	Tanh	SGD	10
80.58	73.11	99.87	84.42	Tanh	SGD	100
97.84	96.47	99.59	98.01	Tanh	RMSprop	10
97.64	96.61	99.06	97.82	Tanh	RMSprop	100

and eventually tend to be stable, as shown by the poly-line in Figure 2. The high-est accuracy rate is 98.27%, and the highest f1-score is 98.39%.

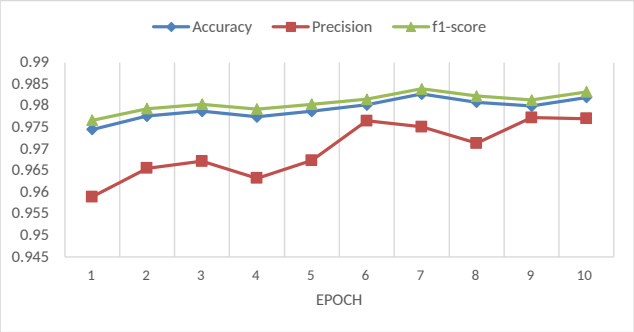


Figure 2: Changes in various indicators during training

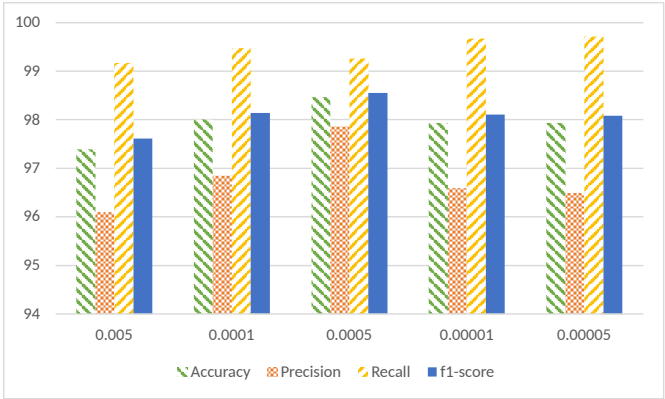


Figure 3: Performance metrics of the model under different learning rates

To further analyze the effect of the learning rate, we selected 6 different learning rates for experiments, and the detailed experimental results are shown in Figure 3. The tracker's performance will suffer if the learning rate is too big or too small. The overall comprehensive performance is optimal when the learning rate is set to 0.0005.

4.4 Influence of Residual Structure on Model Accuracy

This paper compares models with the same number of network layers but no re-sidual structure to verify the efficiency of the residual neural network developed in this paper. The results are shown in Figure 4.

Table 2: Compare with other methods

pape	Method	Accuracy(%)
[10]	Artificial Neural Network & Feature Selection	94.02
[21]	Recurrent Neural Network	97.39
[5]	Convolutional Neural Networks	97.09
[25]	Deep Artificial Neural Networks	97.5
[9]	Random Forest	94.07
Ours	Residual neural network	98.27

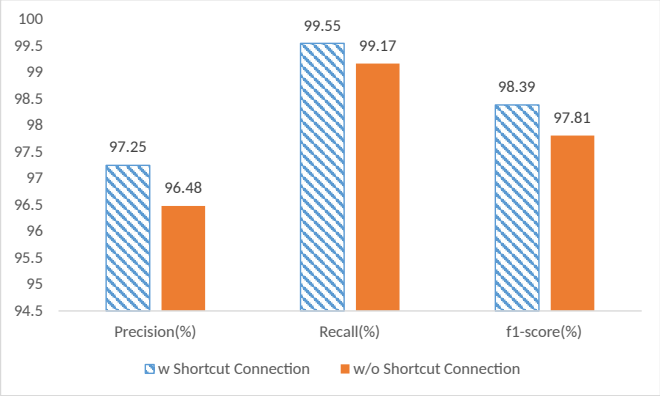


Figure 4: Comparison results of residual neural network and ordinary neural network

When comparing measures like Precision, Recall, and f1-score, it's clear that adding residual structure to the model has significantly improved its performance. Without adding residual structure, the model's Precision, Recall, and f1-score metrics are 96.48%, 99.17%, and 97.81%, respectively. After adding the residual structure, the three indicators of the model reached 97.25%, 99.55%, and 98.39% respectively, which were 0.77%, 0.38%, and 0.58% higher than the former.

4.5 Comparison with Other Models

This study compares the effectiveness of the model provided in this paper to intru-sion detection models created using different methodologies. Table 2 shows the spe-cific outcomes:

When compared to previous intrusion detection models, the intrusion detection model proposed in this paper achieves competitive accuracy results. Compared with the common artificial neural network-based intrusion detection model proposed in [10], it is improved by 4.25%. Compared with the Deep Artificial Neural Networks-based and deep belief network-based models, the improvements are 0.75

5 Conclusion

This paper proposed an intrusion detection model based on residual neural network to address the circumstance where existing intrusion detection models cannot adequately mine the cyberattack elements in traffic information. The residual structure is built in the neural network model using the principle of residual learning, which increases the deep neural network's learning ability. This allows the model to benefit from a deeper network structure while alleviating the dilemma of model overfitting. Finally, the suggested intrusion detection model's performance test and ablation experiments are carried out on the NSL-KDD dataset, and the results demonstrate the proposed model's effectiveness.

Acknowledgments

This study is supported by National Natural Science Foundation of China (62072416), Zhongyuan Science and Technology Innovation Leadership Program (214200510026), and Key Technologies R&D Program of Henan Province (212102210429, 222102210170, 222102210322).

References

[1] Z. Ahmad, A. Shahid Khan, C. Wai Shiang, J. Abdullah, and F. Ahmad, "Network intrusion detection system: A systematic study of machine learning and deep learning approaches," *Transactions on Emerging Telecommunications Technologies*, vol. 32, no. 1, p. e4150, 2021.

[2] M. Ahmed, R. Seraj, and S. M. S. Islam, "The k-means algorithm: A comprehensive survey and performance evaluation," *Electronics*, vol. 9, no. 8, p. 1295, 2020.

[3] R. Chapaneri and S. Shah, "A comprehensive survey of machine learning-based network intrusion detection," *Smart Intelligent Computing and Applications*, pp. 345–356, 2019.

[4] M. A. Ferrag, L. Maglaras, A. Ahmim, M. Derdour, and H. Janicke, "Rdtids: Rules and decision

tree-based intrusion detection system for internet-of-things networks," *Future internet*, vol. 12, no. 3, p. 44, 2020.

[5] N. Gao, L. Gao, Q. Gao, and H. Wang, "An intrusion detection model based on deep belief networks," in *2014 Second International Conference on Advanced Cloud and Big Data*. IEEE, 2014, pp. 247–252.

[6] J. Gu and S. Lu, "An effective intrusion detection approach using svm with naïve bayes feature embedding," *Computers & Security*, vol. 103, p. 102158, 2021.

[7] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[8] T. Hurley, J. E. Perdomo, and A. Perez-Pons, "Hmm-based intrusion detection system for software defined networking," in *2016 15th IEEE International Conference on Machine Learning and Applications (ICMLA)*. IEEE, pp. 617–621, 2016.

[9] M. Iliofotou, H.-c. Kim, M. Faloutsos, M. Mitzenmacher, P. Pappu, and G. Varghese, "Graption: A graph-based p2p traffic classification framework for the internet backbone," *Computer Networks*, vol. 55, no. 8, pp. 1909–1920, 2011.

[10] A. Javaid, Q. Niyaz, W. Sun, and M. Alam, "A deep learning approach for network intrusion detection system," *Eai Endorsed Transactions on Security and Safety*, vol. 3, no. 9, p. e2, 2016.

[11] M. H. Kamarudin, C. Maple, T. Watson, and H. Sofian, "Packet header intrusion detection with binary logistic regression approach in detecting r2l and u2r attacks," in *2015 Fourth International Conference on Cyber Security, Cyber Warfare, and Digital Forensic (CyberSec)*. IEEE, 2015, pp. 101–106.

[12] A. Khraisat, I. Gondal, P. Vamplew, and J. Kamruzzaman, "Survey of intrusion detection systems: techniques, datasets and challenges," *Cybersecurity*, vol. 2, no. 1, pp. 1–22, 2019.

[13] F. Laghrissi, S. Douzi, K. Douzi, and B. Hssina, "Intrusion detection systems using long short-term memory (lstm)," *Journal of Big Data*, vol. 8, no. 1, pp. 1–16, 2021.

[14] H. Liu and B. Lang, "Machine learning and deep learning methods for intrusion detection systems: A survey," *applied sciences*, vol. 9, no. 20, p. 4396, 2019.

[15] H. A. Mahmood, "Network intrusion detection system (nids) in cloud environment based on hidden naïve bayes multiclass classifier," *Al-Mustansiriyah Journal of Science*, vol. 28, no. 2, pp. 134–142, 2017.

[16] M. Mohammadi, T. A. Rashid, S. H. T. Karim, A. H. M. Aldalwie, Q. T. Tho, M. Bidaki, A. M. Rahmani, and M. Hosseinzadeh, "A comprehensive survey and taxonomy of the svm-based intrusion detection systems," *Journal of Network and Computer Applications*, vol. 178, p. 102983, 2021.

[17] S. Pasupathi, V. Shanmuganathan, K. Madasamy, H. R. Yesudhas, and M. Kim, "Trend analysis using agglomerative hierarchical clustering approach for time series big data," *The Journal of Supercomputing*, vol. 77, no. 7, pp. 6505–6524, 2021.

[18] M. Ring, S. Wunderlich, D. Scheuring, D. Landes, and A. Hotho, "A survey of network-based intrusion detection data sets," *Computers & Security*, vol. 86, pp. 147–167, 2019.

[19] N. Shone, T. N. Ngoc, V. D. Phai, and Q. Shi, "A deep learning approach to network intrusion detection," *IEEE transactions on emerging topics in computational intelligence*, vol. 2, no. 1, pp. 41–50, 2018.

[20] K. P. Sinaga and M.-S. Yang, "Unsupervised k-means clustering algorithm," *IEEE access*, vol. 8, pp. 80 716–80 727, 2020.

[21] R. Vinayakumar, M. Alazab, K. Soman, P. Poornachandran, A. Al-Nemrat, and S. Venkatraman, "Deep learning approach for intelligent intrusion detection system," *IEEE Access*, vol. 7, pp. 41 525–41 550, 2019.

[22] Y. Wang, "A multinomial logistic regression modeling approach for anomaly intrusion detection," *Computers & Security*, vol. 24, no. 8, pp. 662–674, 2005.

[23] J. Xu and C. R. Shelton, "Intrusion detection using continuous time bayesian networks," *Journal of Artificial Intelligence Research*, vol. 39, pp. 745–774, 2010.

[24] K. Zeng, M. Ning, Y. Wang, and Y. Guo, "Hierarchical clustering with hard-batch triplet loss for person re-identification," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 13 657–13 665.

[25] J. Zhang, Y. Ling, X. Fu, X. Yang, G. Xiong, and R. Zhang, "Model of the intrusion detection system based on the integration of spatial-temporal features," *Computers & Security*, vol. 89, p. 101681, 2020.

Biography

Zengyu Cai is an associate professor fellow of the College of Computer and Communication Engineering of Zhengzhou University of Light Industry. His research interests include network security and artificial intelligence.

Jingchao Wang is a graduate student fellow of the College of Software Engineering of Zhengzhou University of Light Industry. His research interests include network security and artificial intelligence.

Jianwei Zhang is a professor fellow of the College of Software Engineering of Zhengzhou University of Light Industry. His research interests include next generation network and artificial intelligence.

Yajie Si is a graduate student fellow of the College of Computer and Communication Engineering of Zhengzhou University of Light Industry. Her research interests include network security.

High Capacity Reversible Data Hiding Scheme with Low Distortion Based on Central Prediction

Fang Ren[1,2], Xuemei Yao[1,2], Feiyuan Xue[1], and Zhelin Zhang[1]
(Corresponding author: Xuemei Yao)
School of Cyberspace Security, Xi'an University of Posts and Telecommunications[1]
National Engineering Laboratory for Wireless Security[2]
Xi'an 710121, P.R. China
Email: xuemeiyao1999@163.com
(Received June 24, 2022; Revised and Accepted Oct. 18, 2022; First Online Oct. 22, 2022)

Abstract

This paper proposes a new reversible data hiding scheme based on center prediction and histogram shifting, which aims to increase the embedding capacity with a lower distortion rate. First, the cover image is scanned and divided into blocks, and then the value of the center pixel of each block is taken out to predict other values in the same block to obtain a prediction error histogram. Since the value of the central pixel is close to other values in the same block, the prediction error histogram in our scheme is steeper, and the embedding ability is getting larger. Furthermore, based on the center prediction, we introduce ternary embedding and the selection of multiple peak points into the proposed scheme. Experiments show that the proposed scheme has a lower distortion rate and better performance under the same embedding capacity.

Keywords: Central Prediction; Histogram Shifting; Multi-peak Embedding; Reversible Data Hiding; Ternary Embedding

1 Introduction

In recent years, with the rapid development of the Internet, the security of information transmission has gradually become particularly critical in people's lives. As an important branch of data hiding, reversible data hiding has been widely concerned by researchers. Reversible data hiding refers to embedding secret information into multimedia files in a certain way. With this technology, Users can not only extract the secret information at the extraction end, but also ensure that the original multimedia files are not damaged and completely recovered. At present, reversible data hiding technology has been widely used in military, medical, legal and other fields [18]. Therefore, it is of great theoretical significance and practical value to develop a better reversible data hiding algorithm.

Early reversible data hiding algorithms are based on lossless compression. This type of data hiding method mainly utilizes the redundancy of natural images to compress the carrier image losslessly, leaving room for embedding secret information. For example, Fridrich *et al.* [6] generated the information embedding space by compressing the bit plane with the least redundancy of the carrier image, and then embedded the hash value into the selected bit plane by way of bit replacement. Xuan *et al.* [24] proposed to compress the coefficients of discrete wavelet transform (IWT) with high frequency subbands in order to release space for reversible data hiding. This method has certain effect on increasing the embedding capacity. In 2020, Chang *et al.* [1] proposed a reversible data hiding scheme based on improved locally adaptive coding for compressed images by side match vector quantization (SMVQ). The embedding capacity of the scheme is better than that of many existing reversible data hiding schemes based on compression domain.

Difference expansion is another major algorithm of reversible data hiding techniques, which was first proposed by Tian *et al.* [20]. In this algorithm, the adjacent pixels in the carrier image are divided into multiple pixel pairs, and a data bit is embedded by doubling the difference of each pair of pixel values, which means that the embedding rate of this algorithm can reach 0.5bpp in general. Compared with lossless compression, the performance of difference extension algorithm is improved greatly.

Histogram shifting algorithm was first proposed by Ni *et al.* [14] in 2006. The main idea of this algorithm is to move the pixels between the peak point and the zero point of the gray histogram to leave a spare position, and then embed the data by shifting the peak pixels. The extraction and recovery process is easy to complete as long as the peak and zero points are known at the data extraction end. Compared with the previous algorithms, this algorithm is simpler and its execution time is shorter, the overall performance is also better, which has the representative significance. In the following decades, many improved algorithms based on histogram shifting have been proposed.

Fallahbour and Sedaghi [5] first proposed the application of histogram shifting technology to image blocks, which further improved the overall performance of histogram shifting algorithm. Firstly, the image is divided into blocks, and then the corresponding pixel histogram is generated for each block. Finally, the secret information is embedded through histogram shifting. Lee *et al.* [11] proposed a new reversible data hiding method based on difference histogram. Compared with pixel gray histogram, the histogram of pixel difference has more pixels at the peak point, which means that it can achieve higher embedding capacity. Xuan *et al.* modified the histogram generated by the high-frequency integer wavelet transform coefficients in [23]. In this algorithm, appropriate bin pairs were selected to shift and expand the embedded secret information, so as to minimize the embedding distortion. Subsequently, a general architecture based on histogram shifting was proposed in [12]. According to this framework, users only need to design an image division strategy, an embedding function and a shifting function to create a new reversible data hiding scheme.

Prediction error expansion is one of the most commonly used reversible data hiding algorithms in recent years, which was first proposed by Thodi and Rodriguez [19]. The algorithm uses a pixel predictor to calculate the pixel prediction value and prediction error, and expands the prediction error to embed the secret information. Subsequently, many improved algorithms based on prediction error expansion have appeared [3, 4, 10, 13, 15, 17, 21, 22]. Sachnev [17] proposed a rhombus prediction method, which used the average value of the four pixels closest to the center pixel to predict the center pixel, and adopted a double-layer embedding mechanism to ensure the reversibility of the algorithm.

Luo *et al.* [13] then proposed a pixel prediction method based on interpolation, in which the weighted average of the four pixels closest to the center pixel is used as the prediction value of the center pixel. The prediction result of this method is more accurate. Chen *et al.* [3] proposed a reversible data hiding algorithm based on multiple prediction and asymmetric histogram shifting technology, which can increase the number of pixels at the peak point of the prediction error histogram and reduce the number of shifted pixels. In addition, the algorithm also carries out secondary data embedding, which greatly increases the embedding capacity. Qin *et al.* [15] proposed an assisted prediction method for image inpainting, which uses the image inpainting technology of partial differential equation to realize reversible data hiding. Dragoi and Coltuc [4] proposed a local prediction scheme, which aims to design different predictors for each pixel. Compared with the global prediction method, this scheme can improve the performance of reversible data hiding algorithm effectively. Jung [10] divided the image into 3×1 non-overlapping image blocks, sorted the pixel values in each block, and predicted the maximum and minimum values in the block. Under the condition of low distortion rate, this scheme can embed up to two bits of secret data in every three pixels.

Weng *et al.* [22] improved reversible data hiding algorithm based on pixel values sort. The algorithm firstly divides the image into blocks and sorts the remaining pixels in the block, thenselects three maximum and minimum pixel value points for data embedding. This algorithm can embed up to one bit of data in every two pixels, which further improves the embedding capacity. Huang *et al.* [8] improved the reversible data hiding algorithm based on histogram shifting. After the error histogram is generated, the pixels larger than the value of the peak point are moved to the right. After the data is embedded, the modified error histogram is shifted to the left to achieve double-layer embedding, which further improving the problem of insufficient embedding capacity. In 2022, He *et al.* [7] proposed a reversible data hiding scheme based on multi-predictor and adaptive expansion. The proposed scheme obtains multiple extreme value predictors by extending the prediction context, and combines the given local complexity function to determine the best extreme value predictors for each pixel, which achieves more accurate prediction and outperforms a series of recent schemes.

In addition to reversible data hiding in the plaintext domain, reversible information hiding can be carried out in the encryption domain by combining the above four main methods with encryption technology [2, 9]. The security of reversible data hiding technology can be further enhanced by encrypting the secret information and then embedding it or encrypting the embedded watermarked image.

Based on the results obtained by Ren *et al.* [16], this paper further proposes two new reversible data hiding algorithms based on central prediction, namely ternary embedding and multi-peak embedding, which further increases the embedding capacity on the premise of ensuring good image quality, and verified the effectiveness of the proposed two algorithms through more experimental results. The second part of the paper introduces two reversible data hiding algorithms for subsequent comparison, the third part introduces the proposed scheme in detail, the fourth part compares the experimental results, and the fifth part gives a conclusion.

2 Related Works

In this section, we will introduce Chen's histogram shifting algorithm [3] and Jung's prediction error expansion algorithm based on sorting pixel values [10]. These two algorithms are classical reversible data hiding algorithms and have better performance than most existing algorithms, which can be used for subsequent experimental comparison.

2.1 The Algorithm of Chen

The main idea of Chen *et al.* 's algorithm is to use multiple prediction schemes to predict pixel values, and then use the obtained multiple prediction values of the current pixel to construct the asymmetric prediction error histogram (as shown in Figure 1). Finally, appropriate prediction errors are selected for data embedding.

Compared with the traditional prediction error histogram, the asymmetric prediction error histogram reduces the number of invalid shifted pixels without affecting the data embedding, thus reducing the mean square error caused by the data embedding and improving the visual quality of marked images. In addition, in the process of data embedding, Chen *et al.* proposed a complementary embedding strategy combining the histogram of maximum and minimum prediction error, which greatly increased the embedding capacity through secondary data embedding. After the data extraction is completed, the marked image can be restored to the original image because the two prediction error histograms move in opposite directions respectively during the embedding process.

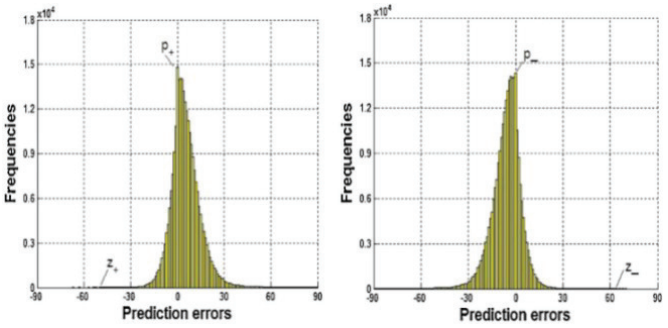


Figure 1: The asymmetric prediction error histograms [3]

2.2 The Algorithm of Jung

Jung proposed a new reversible data hiding method based on sorting pixel values. Firstly, the original image is divided into 1 × 3 non-overlapping image blocks, and the pixel values in the blocks are arranged in ascending order to obtain (x_1, x_2, x_3). Then, the maximum and minimum prediction errors are calculated according to Formula (1). According to the maximum and minimum values of the prediction error, whether the data can be embedded is determined. The specific rules are shown in Formula (2) and (3).

$$\begin{cases} e_{\max} = x_3 - x_2 \\ e_{\min} = x_1 - x_2 \end{cases} \tag{1}$$

$$e'_{\max} = \begin{cases} e_{\max} & if\, e_{\max} = 0 \\ e_{\max} + msg & if\, e_{\max} = 1 \\ e_{\max} + 1 & if\, e_{\max} > 1 \end{cases} \tag{2}$$

$$e'_{\min} = \begin{cases} e_{\min} & if\, e_{\min} = 0 \\ e_{\min} - msg & if\, e_{\min} = -1 \\ e_{\min} - 1 & if\, e_{\min} < -1 \end{cases} \tag{3}$$

The prediction error histogram obtained by this algorithm is steeper, and each 1 × 3 image block can embed up to 2 bits of data. At the same time, the algorithm does not need to embed additional auxiliary information. In the process of data extraction and recovery, the sorting method of pixel value and the calculation method of prediction error are similar to the process of embedding, which can accurately extract secret data and restore the original image.

3 Proposed Algorithm

It can be seen from the second section that the embedding capacity of reversible data hiding algorithm is closely related to the number of pixels at the peak point of the generated histogram. The steeper the generated histogram, the more pixels at the peak point and the larger the embedding capacity. In order to improve the embedding capacity of reversible data hiding algorithm, two histogram shifting algorithms based on central prediction is proposed in this paper. The main ideas of this algorithm are as follows:

1) Firstly, the carrier image is divided into blocks, and the values of surrounding pixels are predicted accurately according to the values of the center pixels of the image blocks. The advantage of this prediction method is that is has higher prediction accuracy in the smooth region of the carrier image, which means that the prediction error histogram generated by the carrier image can gather a large number of pixels at the peak point (as shown in Figure 2), which greatly improves the problem of insufficient embedding capacity;

2) Secondly, in the process of data embedding, we use ternary embedding and multi-peak embedding to control the shifted pixels, thereby reducing the mean square error, improving the value of PSNR, and further increasing the embedding capacity.

3.1 Central Prediction

We propose a center prediction method to predict the pixels in the original carrier image. This prediction method refers to using the center pixel of each image block to predict the values of its adjacent pixels. When the blocks of the image are small enough, the pixel value of the center point of the carrier image is very close to the values of its adjacent pixels. Therefore, using this prediction method can not only obtain a prediction error histogram with a large number of pixels at the point where the prediction error is zero to improve the embedding capacity, but also can make the calculation more concise and save the calculation cost.

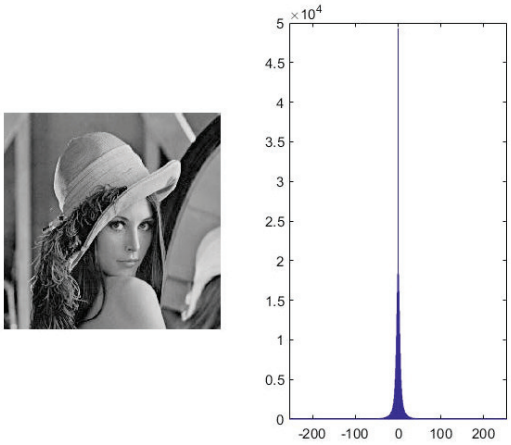
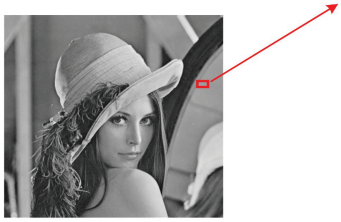


Figure 2: The prediction error histogram of image Lena generated by central prediction



156	156	155
154	**155**	155
156	155	151

(a)

1	1	0
-1		0
1	0	-4

(b)

Figure 3: (a) The image block; (b) The prediction errors

For an image of size M×N (where M is the length of the image and N is the width of the image), we need to divide it into non-overlapping blocks. To ensure that each block has a central pixel point, the size of the block must be $(2m+1)\times(2n+1)$, where m and n are positive integers.

Here, we use $x_{i,j}$ to represent the original pixel value and $x'_{i,j}$ to represent the prediction value. For a pixel block of size $(2m+1)\times(2n+1)$, it is easy to get the gray value of the center point is $med = x_{m+1,n+1}$. Below we will introduce the process of prediction using $m = n = 1$ as an example.

As shown in Figure 3, the gray value of the center point of image block a is extracted to obtain (marked in bold in Figure 3 (a)).

For $(i,j) \neq (2,2)$ in block a, the predicted value is:

$$x'_{i,j} = med. \tag{4}$$

The prediction error is:

$$e_{i,j} = x_{i,j} - x'_{i,j} = x_{i,j} - med, \quad (i,j) \neq (2,2). \tag{5}$$

The prediction results are shown in Figure 3 (b).

3.2 Ternary Embedding Process

We propose two reversible data hiding algorithms based on center prediction, namely ternary embedding and multi-peak embedding. In the specific embedding process of these two algorithms, the prediction error histogram shifting algorithm is used. As mentioned above, compared with other traditional algorithms, the performance of the prediction error histogram shifting algorithm has been greatly improved. This kind of algorithm can not only control the maximum threshold of pixel change by histogram shifting technology, that is, select the peak point in the prediction error histogram of the image to embed the secret information, but also fully consider the correlation between the pixels in the natural image. It has a large embedding capacity while ensuring the visual effect of the watermarked image. The specific steps of the ternary embedding algorithm are as follows:

1) For the given carrier image A and the secret information S, set reasonable values of m and n for dividing A into blocks, and then convert the secret information into elements in the ternary domain, namely:

$$S = s_1, s_2, ..., s_l, \qquad s_i \in \{0,1,2\}. \tag{6}$$

2) According to the distribution of pixels in blocks of original image A, the value of the center pixel in the block is extracted as med, and then the prediction error $e_{i,j}$ can be obtained by Formula (5).

3) After obtaining the prediction error, we will perform different shifting operations on the generated prediction error histogram according to the value of the embedded secret information s_i. In this paper, the original secret information is transformed into the value of the ternary domain for embedding, so we redefine the shifting rules of the prediction error histogram, as described in Formula (7).

$$e'_{i,j} = \begin{cases} e_{i,j} & if\ e_{i,j} = 0\ and\ s_i = 0 \\ e_{i,j} + 1 & if\ e_{i,j} = 0\ and\ s_i = 1 \\ e_{i,j} - 1 & if\ e_{i,j} = 0\ and\ s_i = 2 \\ e_{i,j} + 1 & if\ e_{i,j} > 0 \\ e_{i,j} - 1 & if\ e_{i,j} < 0 \end{cases} \tag{7}$$

4) After the prediction error is modified, the corresponding pixel value of the marked image can be obtained:

$$y_{i,j} = x'_{i,j} + e'_{i,j} = med + e'_{i,j}. \tag{8}$$

Where $y_{i,j}$ is the pixel value of the marked image corresponding to the original carrier image, $e'_{i,j}$ is the modified prediction error.

5) According to Formula (8), the marked values of all pixels in the block are obtained. On this basis, other blocks can be embedded in the same way, finally the marked image is obtained.

Data extraction and recovery is the inverse process of data embedding. Since this algorithm uses the value of the center pixel to predict the value of other pixels, the center pixel is only used for prediction without considering data embedding, so the value of the center pixel will not change in the marked image.The specific steps of the ternary embedding algorithm are as follows:

1) Similar to the processing of the original image block, we first get the value of the center pixel in the marked image block, namely *med*, and the modified prediction error can be calculated by Formula (9).

$$\begin{gathered} e'_{i,j} = y_{i,j} - x'_{i,j} = y_{i,j} - med \\ (i,j) \neq (m+1, n+1) \end{gathered} \tag{9}$$

2) Then sequentially extract the secret data in sequence $e'_{i,j}$, the method of extracting secret data is shown in Formula (10). It should be noted here that the extracted data is ternary data and we need to convert it to binary data after the extraction is complete.

$$s_i = \begin{cases} 0 & if\ e'_{i,j} = 0 \\ 1 & if\ e'_{i,j} = 1 \\ 2 & if\ e'_{i,j} = -1 \end{cases} \tag{10}$$

3) The original image block can be recovered while the secret data is extracted. First, we need to recover the original prediction error:

$$e_{i,j} = \begin{cases} e'_{i,j} & if\ e'_{i,j} = 0 \\ e'_{i,j} - 1 & if\ e'_{i,j} = 1 \\ e'_{i,j} + 1 & if\ e'_{i,j} = -1 \\ e'_{i,j} - 1 & if\ e'_{i,j} > 1 \\ e'_{i,j} + 1 & if\ e'_{i,j} < -1 \end{cases} \tag{11}$$

4) The original pixel value can be calculated according to the original prediction error:

$$x_{i,j} = e_{i,j} + med. \tag{12}$$

5) After the extraction and recovery of the image block is completed, repeating the above steps can complete the extraction and recovery of other blocks, and then the original carrier image can be recovered.

3.3 Multi-peak Embedding Process

The specific steps of the multi-peak embedding algorithm are as follows:

1) Similar to the previous steps of ternary embedding algorithm, for the given carrier image A and the secret information S, set reasonable values of m and n for dividing A into blocks.

2) According to the distribution of pixels in blocks of original image A, the value of the center pixel in the block is extracted as *med*, and then the prediction error $e_{i,j}$ can be obtained by Formula (5).

3) After the prediction error is obtained, we will perform different shifting operations on the generated prediction error histogram according to the value of the embedded secret information. Different from ternary embedding, here we choose two pixel pairs of peak points and zero points for data embedding. The specific embedding rules are shown in Formula (13).

$$e'_{i,j} = \begin{cases} e_{i,j} - 1 & if\ e_{i,j} < \min(p_1, p_2) \\ & and\ e_{i,j} > \min(z_1, z_2) \\ e_{i,j} - b & if\ e_{i,j} = \min(p_1, p_2) \\ e_{i,j} + 1 & if\ e_{i,j} > \max(p_1, p_2) \\ & and\ e_{i,j} < \max(z_1, z_2) \\ e_{i,j} + b & if\ e_{i,j} = \max(p_1, p_2) \\ e_{i,j} & otherwise \end{cases} \tag{13}$$

Where $b \in \{0, 1\}$, representing the binary secret information to be embedded, p_1 and p_2 are the two peak points selected for embedding in the prediction error histogram, z_1 and z_2 are the two zero points corresponding to the two peak points, and $e'_{i,j}$ is the modified prediction error.

4) After the prediction error is modified, the corresponding pixel value of the marked image can be obtained according to Formula (8).

Data extraction and recovery is the inverse process of data embedding. Since both multi-peak point embedding and ternary embedding algorithms adopt the central prediction method, so the value of the center pixel will not change in the marked image. The specific steps of the ternary embedding algorithm are as follows:

1) The modified prediction error can be calculated by Formula (9).Then sequentially extract the secret data in sequence $e'_{i,j}$, The method of extracting the secret data is shown in Formula (14).

$$b = \begin{cases} 1 & if\ e'_{i,j} = \min(p_1, p_2) - 1 \\ & or\ e'_{i,j} = \max(p_1, p_2) + 1 \\ 0 & if\ e'_{i,j} = p_1\ or\ e'_{i,j} = p_2 \end{cases} \tag{14}$$

2) The original image block can be recovered while the secret data is extracted. First, we need to recover the original prediction error:

$$e_{i,j} = \begin{cases} e'_{i,j} + 1 & if\ e'_{i,j} < \min(p_1, p_2) - 1 \\ & and\ e'_{i,j} \geq \min(z_1, z_2) \\ e'_{i,j} + 1 & if\ e'_{i,j} = \min(p_1, p_2) - 1 \\ e'_{i,j} & if\ e'_{i,j} = p_1\ or\ e'_{i,j} = p_2 \\ e'_{i,j} - 1 & if\ e'_{i,j} > \max(p_1, p_2) + 1 \\ & and\ e'_{i,j} \leq \max(z_1, z_2) \\ e'_{i,j} - 1 & if\ e'_{i,j} = \max(p_1, p_2) + 1 \\ e'_{i,j} & otherwise \end{cases} \tag{15}$$

Where the peak points p_1 and p_2, the zero points z_1 and z_2 are the same as that in the embedding process.

3) Then the original pixel value can be calculated according to Formula (12), and the original carrier image block can be recovered. After the extraction and recovery of the image block is completed, repeating the above steps can complete the extraction and recovery of other blocks, and then the original carrier image can be recovered.

3.4 Dealing with Overflow and Underflow

Due to the problem of overflow or underflow in the shifting process of prediction error histogram, some pixels in the original image cannot be reversibly recovered, so we need to do some preprocessing for the original carrier image. The preprocessing is mainly aimed at the points whose original pixel value is 0 and needs to move to the left during embedding, or the points whose original pixel value is 255 and need to move to the right during embedding. We need to record these points, which are not used for embedding, but their location information needs to be embedded as additional information along with the secret information. These points are excluded when the prediction error histogram is shifted.

3.5 Analysis of Time Efficiency and Space Efficiency

Reversible data hiding technology needs to complete the embedding process of secret information in a very short time, otherwise it will lose its practical application significance due to the high computational cost. Through reading literatures and experiments, it can be found that most of the reversible data hiding schemes, including the two algorithms proposed in this paper and the four comparison algorithms, can complete the embedding process of secret information in about 1 second, and there is no significant difference in running time. Therefore, here we mainly make a brief analysis of the proposed algorithm from the two aspects of time complexity and space complexity.

Assuming that the size of the original image is M×N, the space complexity of the two proposed algorithms mainly consists of three aspects: reading the original image, storing the prediction error and generating the watermarked image. Therefore, the space complexity is $O(3MN)$, which occupies less system memory space.

In addition, the time complexity of the two proposed algorithms also consists of three stages: original image segmentation, prediction error calculation, and prediction error histogram shifting. In the stage of original image segmentation, we divide the original image into 3×3 image blocks, and the time complexity of this stage is $O(MN)$. In the stage of prediction error calculation, we need to calculate the differences between the pixel value of the center point of each image block and the remaining pixels in the block, the time complexity of this stage is $O(MN)$. In the stage of prediction error histogram shifting, we need to shift the pixels between the peak point and the zero point to the left or right by one unit, that is, the pixel gray value is increased or decreased by one, the time complexity of this stage is $O(kMN)$, where k is less than one-half. So for the two proposed algorithms, the time complexity of each algorithm is $O(MN)$, which indicates that we can get the experimental results in the linear time of the image size, that is, the two algorithms are feasible.

4 Experimental Results and Analysis

In this section, we implement the two proposed algorithm and four comparison algorithms by programming, and evaluate the performance of the proposed algorithm from two aspects of peak signal-to-noise ratio(PSNR) and embedding capability by analyzing the experimental results.

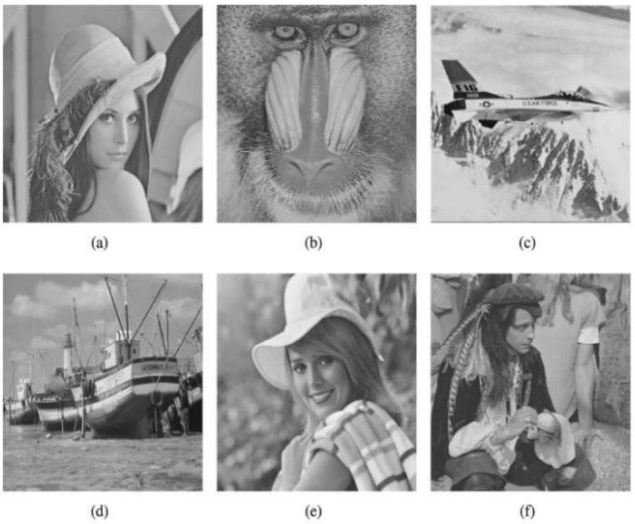


Figure 4: Test images: Lena, Baboon, Airplane, Boat, Elaine and Man

First of all, six standard grayscale images (Lena, Baboon, Airport, Boat, Elaine and Man) with a size of 512×512 are selected from USC-SIPI image database (as shown in Figure 4) for testing. The experimental results are compared with Jung's pixel value sorting algorithm [10], Weng's improved pixel value sorting algorithm [22], Huang's improved error histogram shifting algorithm [8] and He's multi-predictor algorithm [7]. Secondly, in order to reduce the impact of the randomness of the selection of test images on the experimental results, we also select 30 grayscale images from the library to test the average results.

4.1 Comparison of Embedding Capacity

Since the six algorithms need to embed data at the peak point of prediction error histogram, the number of bits that can embed data is equal to the number of pixels at the peak point without considering additional information. Here, the maximum embedding rate is used as a metric to evaluate the embedding capability of each algorithm, that is, how many bits of data can be embedded per pixel at most. The formula for calculating the embedding rate of 512×512 test images is as follows:

$$EC = \frac{p_{\max}}{512 \times 512}. \tag{16}$$

Where $p_{\max}$ is the number of pixels of peak points in the prediction error histogram.

It should be noted here that the data extracted from the ternary embedding algorithm in this paper are all ternary data, so the extracted data should be converted into binary data before calculating the maximum embedding rate of the ternary embedding algorithm. The maximum embedding rate after transformation is shown in Table 1.

Table 1: The maximum embedding rate of ternary embedding algorithm

Test image	$EC_{ternary}$	$EC = EC_{ternary} \cdot \log_2 3$
Lena	0.18	0.29
Baboon	0.13	0.21
Airport	0.23	0.37
Boat	0.19	0.30
Elaine	0.16	0.25
Man	0.17	0.27

Figure 5 shows the maximum embedding capacity of the two proposed algorithms and the other four comparison algorithms in six test images. For the test image Lena with relatively smooth texture, the ternary embedding algorithm proposed in this paper can embed 76,021 bits of data at most, and the maximum embedding rate of this algorithm is as high as 0.29, which is 0.19, 0.23, 0.07 and 0.06 higher than Jung's pixel value sorting algorithm [10], Weng's improved pixel value sorting algorithm [22], Huang's improved error histogram shifting algorithm [8] and He's multi-predictor algorithm [7] respectively. The proposed multi-peak embedding algorithm can embed 62914 bits of data in Lena at most, and the embedding rate is as high as 0.24, which is 0.14, 0.18, 0.02 and 0.01 higher than Jung's [10], Weng's [22], Huang's [8] and He's [7] algorithms respectively. For the test image Baboon with relatively rough texture, the maximum embedding rate of the proposed ternary embedding algorithm is 0.21, and that of the multi-peak embedding algorithm is 0.15, which is still higher than the other four comparison algorithms.

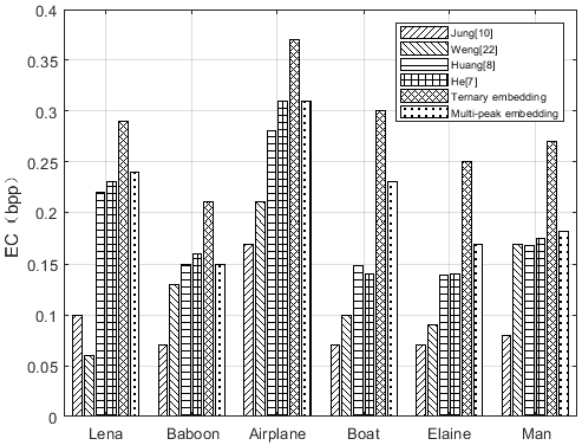


Figure 5: Maximum embedding rate

It can be seen intuitively from Figure 5 that the two algorithms proposed in this paper effectively improve the embedding capability to varying degrees. This is because the center prediction method is used in this paper, that is, the value of the center pixel of the image block is used to predict the value of its surrounding pixels. This method produces more accurate predictions and more pixels at the peak point. In addition, it should be noted that compared with Jung's, Weng's and He's algorithms, there is a smaller gap between Huang's algorithm and the two algorithms proposed in this paper. This is because the double-layer embedding mechanism is used in Huang's algorithm to improve the embedding capability.

4.2 Comparison of PSNR

Peak signal-to-noise ratio (PSNR) is usually used as an index to measure the image quality, which reflects the similarity between the original carrier image and the marked image. The higher the value of PSNR, the more similar the marked image is to the original carrier image, and vice versa. When calculating the PSNR of an image, the mean square error (MSE) of the image must be calculated first. For a test image with a size of 512×512, the mean square error (MSE) is defined as:

$$MSE = \frac{1}{512 \times 512} \sum_{i=1}^{512} \sum_{j=1}^{512} (x_{i,j} - y_{i,j})^2. \tag{17}$$

Where $x_{i,j}$ is the pixel value of the original image and $y_{i,j}$ is the pixel value of the marked image.

The calculation formula of PSNR is as follows:

$$PSNR = 10 \times \log_{10} \frac{255^2}{MSE}. \tag{18}$$

Figure 6 (6(a)-6(f)) shows the comparison of PSNR of each algorithm at the same embedding rate. Here for different test images, this paper sets the initial embedding

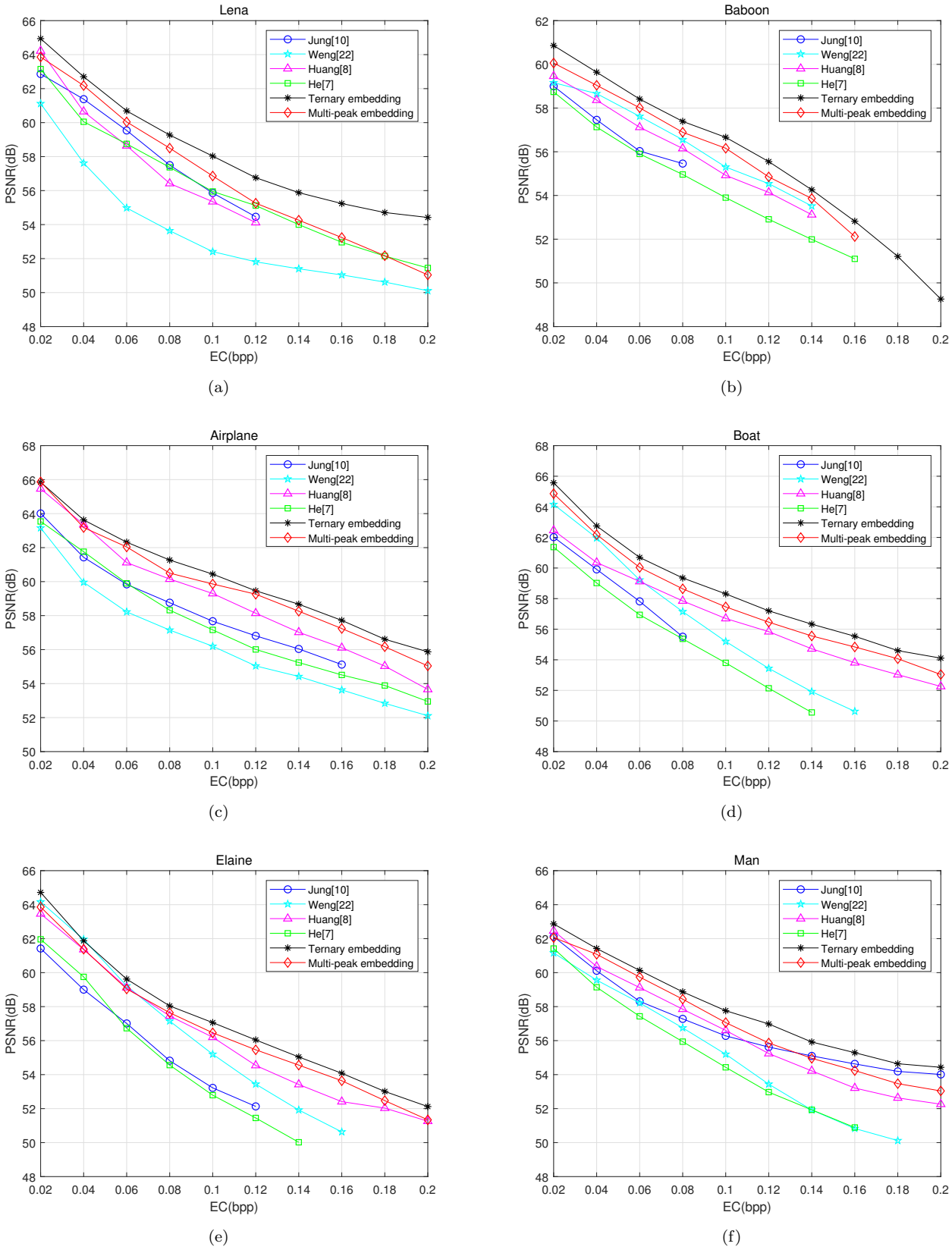


Figure 6: Comparison of PSNR at the same embedding rate

Figure 7: Test images

Table 2: The maximum embedding rate of ternary embedding algorithm and multi-peak embedding algorithm (bpp)

Image	ternary embedding	multi-peak embedding	Image	ternary embedding	multi-peak embedding	Image	ternary embedding	multi-peak embedding
Image01	0.25	0.18	Image11	0.11	0.14	Image21	0.43	0.39
Image02	0.12	0.17	Image12	0.23	0.21	Image22	0.50	0.41
Image03	0.44	0.40	Image13	0.16	0.16	Image23	0.43	0.43
Image04	0.25	0.24	Image14	0.24	0.24	Image24	0.55	0.49
Image05	0.22	0.20	Image15	0.19	0.16	Image25	0.43	0.38
Image06	0.41	0.40	Image16	0.36	0.36	Image26	0.24	0.22
Image07	0.42	0.31	Image17	0.21	0.24	Image27	0.30	0.24
Image08	0.11	0.21	Image18	0.41	0.42	Image28	0.18	0.13
Image09	0.48	0.48	Image19	0.30	0.31	Image29	0.23	0.31
Image10	0.22	0.33	Image20	0.50	0.47	Image30	0.26	0.27

Table 3: PSNR of ternary embedding algorithm and multi-peak embedding algorithm at 0.1bpp embedding rate (dB)

Image	ternary embedding	multi-peak embedding	Image	ternary embedding	multi-peak embedding	Image	ternary embedding	multi-peak embedding
Image01	52.94	51.42	Image11	52.66	55.46	Image21	60.12	59.58
Image02	52.14	52.14	Image12	50.67	50.97	Image22	56.19	54.99
Image03	52.64	51.44	Image13	52.47	52.17	Image23	58.43	57.20
Image04	54.52	53.42	Image14	51.65	52.65	Image24	57.94	57.94
Image05	53.31	52.41	Image15	57.92	56.82	Image25	53.21	53.03
Image06	55.91	55.90	Image16	52.23	52.13	Image26	55.42	55.42
Image07	55.94	56.45	Image17	61.20	60.21	Image27	51.36	52.06
Image08	49.15	49.95	Image18	55.93	55.93	Image28	58.98	57.58
Image09	58.46	56.89	Image19	58.26	58.46	Image29	54.90	54.80
Image10	50.92	50.69	Image20	61.32	60.11	Image30	51.25	52.05

rate to 0.02bpp and increments it in steps of 0.02bpp. For the test image Baboon with relatively rough texture, when the embedding rate is 0.02, the PSNR of the proposed ternary embedding algorithm is 60.86dB, which is 1.86dB, 1.70dB, 1.40dB and 2.12dB higher than Jung's pixel value sorting algorithm [10], Weng's improved pixel value sorting algorithm [22], Huang's improved error histogram shifting algorithm [8] and He's multi-predictor algorithm [7] respectively. The PSNR of the proposed multi-peak algorithm is 60.06dB, which is 1.06dB, 0.90dB, 0.60dB and 1.32dB higher than Jung's [10], Weng's [22], Huang's [8] and He's [7] algorithms respectively. For the test image Airplane with relatively smooth texture, the PSNR of the six algorithms are very high, but the two algorithms proposed in this paper are still superior to the other four comparison algorithms. For example, when the embedding rate is 0.02, the PSNR of the proposed ternary embedding algorithm can reach 65.85dB, and the PSNR of the proposed multi-peak embedding algorithm can reach 65.87dB.

It can be seen from Figure 6 that the performance of the two algorithms proposed in this paper is significantly higher than the other four comparison algorithms under the same embedding rate. This is not only because the central prediction method is used in this paper to obtain more accurate prediction value, but also because the introduction of ternary embedding and multi-peak embedding makes the proposed algorithm reduce invalid shifted pixels at the same embedding rate, thereby reducing the mean square error and improving the value of PSNR.

In addition, in order to reduce the influence of randomness of test image selection on experimental results and further verify the performance improvement of the two algorithms proposed, we select 30 grayscale images with a size of 512×512 (as shown in Figure 7) from USC-SIPI image database for more tests. The specific experimental data are shown in Table 2 and Table 3. The maximum embedding rates of ternary embedding algorithm and multi-peak embedding algorithm are shown in Table 2. When the embedding rate is 0.1bpp, the corresponding PSNR values of ternary embedding algorithm and multi-peak embedding algorithm are shown in Table 3.

As can be seen from Table 2, the average maximum embedding rate of the ternary embedding algorithm proposed in this paper is 0.306bpp, and that of the multi-peak embedding algorithm is 0.297bpp. As can be seen from Table 3, when the embedding rate of secret information is 0.1bpp, the average PSNR of ternary embedding algorithm is 54.93dB, and that of multi-peak embedding algorithm is 54.68dB. The above average results are similar to the experimental results of the six classical grayscale images in Figure 6, which shows that the proposed two algorithms have improved image embedding capacity and visual quality stably.

5 Conclusions

In order to improve the embedding capacity and visual quality of marked images in reversible data hiding scheme, two reversible data hiding schemes based on central prediction are proposed in this paper. In the proposed two schemes, the carrier image is firstly divided into many image blocks, and then the pixel value of the center point of the image block remains unchanged and the value of the adjacent pixels is predicted by this point. Finally, ternary embedding and multi-peak embedding are used to shift the generated prediction error histogram to embed secret information. Since the value of the center pixel of the image block is closer to those of the surrounding pixels, the prediction error histogram generated by the central prediction method is very steep, that is, higher embedding capacity can be achieved. In addition, when embedding secret information, ternary and multi-peak embedding algorithms are introduced in this paper respectively, which further reduce invalid shifted pixels in the histogram shifting process at the same embedding rate, effectively improve the PSNR of marked images, and well improve the visual quality of marked images. Experimental results show that the two proposed reversible data hiding algorithms based on central prediction are superior to other algorithms in terms of embedding capacity and PSNR.

Acknowledgments

This work was supported by the National Nature Science Foundation of China [grant number 61902315, 61802243], the Natural Science Basic Research Plan in Shaanxi Province of China [grant number 2021JM-463] and the Graduate Innovation Fund of Xi'an University of Posts and Telecommunications [grant number CXJJLY202021].The authors gratefully acknowledge the anonymous reviewers for their valuable comments.

References

[1] C. C. Chang, J. Y. Chen, Y. H. Chen, and Y. J. Liu, "A reversible data hiding method for smvq indices based on improved locally adaptive coding," *International Journal of Network Security*, vol. 22, no. 4, pp. 575–583, 2020.

[2] H. Chen, C. C. Chang, and K. Chen, "Reversible data hiding schemes in encrypted images based on the paillier cryptosystem," *International Journal of Network Security*, vol. 22, no. 3, pp. 523–533, 2020.

[3] X. Chen, X. Sun, H. Sun, Z. Zhou, and J. Zhang, "Reversible watermarking method based on asymmetric-histogram shifting of prediction errors," *Journal of Systems and Software*, vol. 86, no. 10, pp. 2620–2626, 2013.

[4] I. C. Dragoi and D. Coltuc, "Local-prediction-based difference expansion reversible watermarking," *IEEE*

Transactions on image processing, vol. 23, no. 4, pp. 1779–1790, 2014.

[5] M. Fallahpour and M. H. Sedaaghi, "High capacity lossless data hiding based on histogram modification," *Ieice Electron Express*, vol. 4, no. 7, pp. 205–210, 2007.

[6] J. Fridrich, M. Goljan, and R. Du, "Invertible authentication," in *Security and Watermarking of Multimedia contents III*, vol. 4314, pp. 197–209, 2001.

[7] W. He, G. Xiong, and Y. Wang, "Reversible data hiding based on multi-predictor and adaptive expansion," *IET Image Processing*, vol. 16, no. 3, pp. 888–899, 2022.

[8] D. Huang and J. Wang, "Efficient reversible data hiding based on the histogram modification of differences of pixel differences," *Multimedia Tools and Applications*, vol. 79, no. 29, pp. 20881–20896, 2020.

[9] B. Jana, "Dual image based reversible data hiding scheme using weighted matrix," *International Journal of Electronics and Information Engineering*, vol. 5, no. 1, pp. 6–19, 2016.

[10] K. H. Jung, "A high-capacity reversible data hiding scheme based on sorting and prediction in digital images," *Multimedia Tools and Applications*, vol. 76, no. 11, pp. 13127–13137, 2017.

[11] S. K. Lee, Y. H. Suh, and Y. S. Ho, "Reversiblee image authentication based on watermarking," in *IEEE International Conference on Multimedia and Expo*, pp. 1321–1324, Toronto, 2006.

[12] X. Li, B. Li, B. Yang, and T. Zeng, "General framework to histogram-shifting-based reversible data hiding," *IEEE Transactions on Image Processing*, vol. 22, no. 6, pp. 2181–2191, 2013.

[13] L. Luo, Z. Chen, C. Ming, Z. Xiao, and X. Zhang, "Reversible image watermarking using interpolation technique," *IEEE Transactions on Information Forensics and Security*, vol. 5, no. 1, pp. 187–193, 2010.

[14] Z. Ni, Y. Q. Shi, N. Ansari, and S. Wei, "Reversible data hiding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 16, no. 3, pp. 354–362, 2006.

[15] C. Qin, C. C. Chang, Y. H. Huang, and L. T. Liao, "An inpainting-assisted reversible steganographic scheme using a histogram shifting mechanism," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 23, no. 7, pp. 1109–1118, 2013.

[16] F. Ren, F. Y. Xue, and X. M. Yao, "High capacity reversible data hiding algorithm based on central prediction," in *2020 International Conference on Networking and Network Applications (NaNA)*, pp. 361–367, 2020.

[17] V. Sachnev, H. J. Kim, J. Nam, S. Suresh, and Q. S. Yun, "Reversible watermarking algorithm using sorting and prediction," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 19, no. 7, pp. 989–999, 2009.

[18] Y. Q. Shi, X. Li, X. Zhang, H. T. Wu, and B. Ma, "Reversible data hiding: advances in the past two decades," *IEEE Access*, vol. 4, pp. 3210–3237, 2016.

[19] D. M. Thodi and J. J. Rodríguez, "Expansion embedding techniques for reversible watermarking," *IEEE transactions on image processing*, vol. 16, no. 3, pp. 721–730, 2007.

[20] J. Tian, "Reversible data embedding using a difference expansion," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 8, pp. 890–896, 2003.

[21] J. Wang, J. Ni, X. Zhang, and Y. Q. Shi, "Rate and distortion optimization for reversible data hiding using multiple histogram shifting," *IEEE transactions on cybernetics*, vol. 47, no. 2, pp. 315–326, 2017.

[22] S. Weng, Y. Chen, B. Ou, C. C. Chang, and C. Zhang, "Improved k-pass pixel value ordering based data hiding," *IEEE Access*, vol. 7, pp. 34570–34582, 2019.

[23] G. Xuan, Y. Q. Shi, P. Chai, C. Xia, Z. Ni, and X. Tong, "Optimum histogram pair based image lossless data embedding," in *International Workshop on Digital Watermarking*, pp. 264–278, Springer, Berlin, Heidelberg, 2008.

[24] G. Xuan, J. Zhu, J. Chen, Y. Q. Shi, Z. Ni, and W. Su, ""distortionless data hiding based on integer wavelet transform,"," *Electronics Letters*, vol. 38, pp. 1646–1648, 2003.

Biography

Fang Ren received the Ph.D. degree in cryptography from Xidian University in 2012. Now he is an associate professor of Xi'an University of Posts and Telecommunications. His current research interests include information security and code-based cryptography.

Xuemei Yao received the B.S. degree from Xi'an University of Posts and Telecommunications in 2020. She is currently going in for the M.S. degree in information security with Xi'an University of Posts and Telecommunications. Her research interests concentrate on reversible data hiding and image security.

Feiyuan Xue received the M.S. degree from Xi'an University of Posts and Telecommunications in 2022. His research interests concentrate on network and information security.

Zhelin Zhang received the B.S. degree from Xi'an University of Posts and Telecommunications in 2020. She is currently going in for the M.S. degree in information security with Xi'an University of Posts and Telecommunications. Her research interests concentrate on data hiding and cryptography.

Reviewers (Volume 24, 2022)

Dariush Abbasinezhad
Tarek Abbes
Ahmed Abd El-Rahiem Abd El-Latif
Slim Abdelhedi
Mohd Faizal Abdollah
Ahmed Mohammed Abdullah
Subrata Acharya
Sodeif Ahadpou
Tohari Ahmad
Muhammad Najmi Ahmad-Zabidi
Mohammad Reza Ahmadi
Asimi Ahmed
Ganesh V. Aithal
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
Seth Alornyo
Ali Mohammed Alsahlany
Richard Amankwah
Ruhul Amin
Rengarajan Amirtharajan
R. Anand
Karl Andersson
Benjamin Arazi
K. S. Arvind
Muhammad Asad
Travis Atkison
Hany Fathy Atlam
Cossi Blaise Avoussoukpo
Anant M. Bagade
Amandeep Bagga
Nazrulazhar Bahaman
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
Dharmendra Bhatti
Krishna Bhowal
Li Bin
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
Chin-Ling Chen
Jan Min Chen
Tzung-Her Chen
Xi Chen
Yang Chen
Yi-Hui Chen
Yushuang Chen
Zhixiong 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
Mooramreddy Sree Devi
Sankhanil Dey
Subhasish Dhal
Jintai Ding
Jingnan Dong
Xiaoli Dong
Nishant Doshi
Ahmed Drissi
Crystal Wilson Dsouza
Qi Duan
Ashraf Diaa Elbayoumy
Abd Allah Adel Elhabshy
Ahmed A. Elngar
Edwin Engin Yaz
Aoxiong Fan
Arizona Firdonsyah
Xingbing Fu
Vladimir Sergeevich Galyaev
Rakesh C Gangwar
Juntao Gao
Tiegang Gao
Xinwei Gao
N. B. Gayathri
G. Geetha
Mohammad GhasemiGol
Madhumala Ghosh
Ramesh Gopalan
Poornima Ediga Goud
Krishan Kumar Goyal
Ke Gu
Avinash k Gulve
Sumalatha Gunnala
Shuai Guo
C. P. Gupta
Jatin Gupta
Pynbianglut Hadem
Charifa Hanin
Ali Hassan
Wien Hong
Tsung-Chih Hsiao
Chengyu Hu
Defa Hu
Xiong Hu
Yen-Hung Hu
Huajun Huang
Chin-Tser Huang
Jianmeng Huang
Munawar Hussain
Bala Venkateswarlu Isunuri
Grasha Jacob
Amit Jain
Yogendra Kumar Jain
Swati Jaiswal
Teena Jaiswal
Bappaditya Jana
V. S. Janani
N Jeyanthi
lin zhi jiang
Shaoquan Jiang
Rong Jiang
Rui Jiang
Zhengping Jin
Ashish Joshi
Li Su Juan
Omprakash Kaiwartya
Yoshito Kanamori
Nirmalya Kar
Gagandeep Kaur
Wongyos Keardsri
Omar Khadir
Vaishali D. Khairnar
Asif Uddin Khan
Md. Al-Amin Khandaker
Malik Sikander Hayat Khiyal
Dong Seong Kim
Kingsford Kissi Mireku
Vikas K Kolekar
P. Dhandapani Raman D. Kothandaraman
Anjan Krishnamurthy
Fengfei Kuang
Sajja Ratan Kumar
Manish Kumar
Naresh N Kumar
Saru Kumari
Yesem Kurt Peker
Owusu-Agyemang Kwabena
Albert Kofi Kwansah Ansah
Manmohan Lakhera
Then Lee
Cheng Li
Chun-Ta Li
Yanping Li
Zhaozheng Li
H. M. Lian
Changlu Lin

Chia-Chen Lin
Chih-Yang Lin
Iuon-Chang Lin
Yang-Bin Lin
Jiang Hong Ling
Desheng Liu
Li Liu
Shuang Gen Liu
Ting Liu
Ximeng Liu
Yanjun Liu
Yining Liu
K. Shantha Kumari Luke
Jayakumar
Zhiyong Luo
Ming Luo
Sagar Bhaskar Mahajan
Zahid Mahmood
Tanmoy Maitra
Doaa Mohsin Majeed
Arun Malik
Mahalinga V. Mandi
T. Manesh
Palvinder Singh Mann
Ali Mansouri
A. M. Meddeb-Makhlouf
Kamran Ali Memon
Bo Meng
Weizhi Meng
Yang Ming
Suhail Qadir Mir
Amit Mishra
Anuranjan Misra
Syed Shahul Hameed
Mohamed Ismail
Sirwan Ahmed Mohammed
Madihah Mohd Saudi
Guillermo Morales-Luna
Belmekki Mostafa
Alaa Moualla
Hamdy M. Mousa
Muhammad M. Muhammad
Kuntal Mukherjee
C. H. Mukundha
Bhagavathi Priya M
Muthumanikam
Ambika Nagaraj
Preeti Nagrath
K. Nandhini
Syed Naqvi
Kanagaraj Narayanasamy
Lakshmi Kannan Narnayanan
Prabir Kr Naskar
Sarmistha Neogy
Krishnamur G Ningappa
Sohail Noman
Chokri Nouar
Abdul Abiodun Orunsolu
Arezou Ostad Sharif
Nasrollah Pakniat
Dhiraj Pandey
S. K. Pandey
B. D. Parameshachari
Subhash S. Parimalla
Chintan J. Patel
Kailas Ravsaheb Patil
Suresh Kumar Peddoju
Hongmei Pei
Kanthakumar Pongaliur
A. Prakash
Krishna K. Prakash
Munivara Prasad
Hongquan Pu
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
Ganga Rama Koteswara Rao
Golagani A.V.R.C Rao
Mohammad Maher Rasheed
V. Rathinasabapathy
Dhivya Ravi
Ramesh S Rawat
Siva Ranjani Reddi
Khaled Riad
Mohd Foad Rohani
Ou Ruan
Sanjay Kumar Sahay
Ashish Saini
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
Abhishek Shukla
Varun Shukla
Anuj Kumar Singh
Debabrata Singh
Jitendra Singh
Mahendra Pratap Singh
Mukesh Singh
Bala Srinivasan
Siva Shankar Subramanian
Karthikeyan Subramanian
T. SudalaiMuthu
K. S. Suganya
Guodong Su
Haiyan Sun
Fei Tang
Maryam Tanha
Ariel Soares Teles
Pratik Teli
Xiuxia Tian
Geetam Singh Tomar
Yuan-Yu Tsai
Pushpendra Kumar Verma
Ravi Verma
Vandani Verma
Vibhor Kumar Vishnoi
Phu Vo Ngoc
Tao Wan
Putra Wanda
Ding Wang
Fangwei Wang
Feng Wang
Guoqing Wang
Li Wang
Libin Wang
Linfan Wang
Qingping Wang
Xiaogang Wang
Xingbo Wang
Xu Wang
Ying Wang
C. H. Wei
Jianghong Wei
Zhe Wei
Axin Wu
Na-I Wu
Chengbo Xu
Degang Xu
Lei Xu
Chengbo Xu
Yashveer Yadav
Wei Yajuan
Jun Yan
Changsong Yang
Li Yang
Rui Yang
Wenjie Yang
Yifei Yao
Jun Ye
Pinghao Ye
Fangfang Yin
Huang Yiwang
Lin You
Huifang Yu
Lei Yu
Hang Yue
Taskeen Zaidi
Noor Zaman Zaman
Jianjun Zhang
Sherali Zeadally
Jianping Zeng
Fangguo Zhang
Futai Zhang
Jianhong Zhang
Jie Xiu Zhang
Qiu-Yu Zhang
Shanshan Zhang
Yanshuo Zhang
Yinghui Zhang
Zonghua Zhang
Hongzhuan Zhao
Mingju Zhao
Yuntao Zhao
Zhiping Zhou
Ye Zhu
Yingwu Zhu
Frank Zhu
Aaron Zimba

Guide for Authors
International Journal of Network Security

IJNS will be committed to the timely publication of very high-quality, peer-reviewed, original papers that advance the state-of-the art and applications of network security. Topics will include, but not be limited to, the following: Biometric Security, Communications and Networks Security, Cryptography, Database Security, Electronic Commerce Security, Multimedia Security, System Security, etc.

1. Submission Procedure

Authors are strongly encouraged to submit their papers electronically by using online manuscript submission at http://ijns.jalaxy.com.tw/.

2. General

Articles must be written in good English. Submission of an article implies that the work described has not been published previously, that it is not under consideration for publication elsewhere. It will not be published elsewhere in the same form, in English or in any other language, without the written consent of the Publisher.

2.1 Length Limitation:

All papers should be concisely written and be no longer than 30 double-spaced pages (12-point font, approximately 26 lines/page) including figures.

2.2 Title page

The title page should contain the article title, author(s) names and affiliations, address, an abstract not exceeding 100 words, and a list of three to five keywords.

2.3 Corresponding author

Clearly indicate who is willing to handle correspondence at all stages of refereeing and publication. Ensure that telephone and fax numbers (with country and area code) are provided in addition to the e-mail address and the complete postal address.

2.4 References

References should be listed alphabetically, in the same way as follows:
For a paper in a journal: M. S. Hwang, C. C. Chang, and K. F. Hwang, ``An ElGamal-like cryptosystem for enciphering large messages," *IEEE Transactions on Knowledge and Data Engineering*, vol. 14, no. 2, pp. 445--446, 2002.

For a book: Dorothy E. R. Denning, *Cryptography and Data Security*. Massachusetts: Addison-Wesley, 1982.

For a paper in a proceeding: M. S. Hwang, C. C. Lee, and Y. L. Tang, ``Two simple batch verifying multiple digital signatures," in *The Third International Conference on Information and Communication Security (ICICS2001)*, pp. 13--16, Xian, China, 2001.

In text, references should be indicated by [number].

Subscription Information

Individual subscriptions to IJNS are available at the annual rate of US$ 200.00 or NT 7,000 (Taiwan). The rate is US$1000.00 or NT 30,000 (Taiwan) for institutional subscriptions. Price includes surface postage, packing and handling charges worldwide. Please make your payment payable to "Jalaxy Technique Co., LTD." For detailed information, please refer to http://ijns.jalaxy.com.tw or Email to ijns.publishing@gmail.com.