

Data Security and Privacy for Fog/ Edge Computing-Based IoT

Lead Guest Editor: Jie Cui

Guest Editors: Antonio Liotta, Ke Gu, and Lu Liu





Data Security and Privacy for Fog/Edge Computing-Based IoT

Security and Communication Networks

Data Security and Privacy for Fog/Edge Computing-Based IoT

Lead Guest Editor: Jie Cui

Guest Editors: Antonio Liotta, Ke Gu, and Lu Liu






Copyright © 2022 Hindawi Limited. All rights reserved.

This is a special issue published in "Security and Communication Networks." All articles are open access articles distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Chief Editor

Roberto Di Pietro, Saudi Arabia

Associate Editors

Jiankun Hu , Australia
Emanuele Maiorana , Italy
David Megias , Spain
Zheng Yan , China

Academic Editors








Saed Saleh Al Rabae , United Arab Emirates
Shadab Alam, Saudi Arabia
Goutham Reddy Alavalapati , USA
Jehad Ali , Republic of Korea
Jehad Ali, Saint Vincent and the Grenadines
Benjamin Aziz , United Kingdom
Taimur Bakhshi , United Kingdom
Spiridon Bakiras , Qatar
Musa Balta, Turkey
Jin Wook Byun , Republic of Korea
Bruno Carpentieri , Italy
Luigi Catuogno , Italy
Ricardo Chaves , Portugal
Chien-Ming Chen , China
Tom Chen , United Kingdom
Stelvio Cimato , Italy
Vincenzo Conti , Italy
Luigi Coppolino , Italy
Salvatore D'Antonio , Italy
Juhriyansyah Dalle, Indonesia
Alfredo De Santis, Italy
Angel M. Del Rey , Spain
Roberto Di Pietro , France
Wenxiu Ding , China
Nicola Dragoni , Denmark
Wei Feng , China
Carmen Fernandez-Gago, Spain
AnMin Fu , China
Clemente Galdi , Italy
Dimitrios Geneiatakis , Italy
Muhammad A. Gondal , Oman
Francesco Gringoli , Italy
Biao Han , China
Jinguang Han , China
Khizar Hayat, Oman
Azeem Irshad, Pakistan

M.A. Jabbar , India
Minho Jo , Republic of Korea
Arijit Karati , Taiwan
ASM Kayes , Australia
Farrukh Aslam Khan , Saudi Arabia
Fazlullah Khan , Pakistan
Kiseon Kim , Republic of Korea
Mehmet Zeki Konyar, Turkey
Sanjeev Kumar, USA
Hyun Kwon, Republic of Korea
Maryline Laurent , France
Jegatha Deborah Lazarus , India
Huaizhi Li , USA
Jiguo Li , China
Xueqin Liang, Finland
Zhe Liu, Canada
Guangchi Liu , USA
Flavio Lombardi , Italy
Yang Lu, China
Vincente Martin, Spain
Weizhi Meng , Denmark
Andrea Michienzi , Italy
Laura Mongioi , Italy
Raul Monroy , Mexico
Naghme Moradpoor , United Kingdom
Leonardo Mostarda , Italy
Mohamed Nassar , Lebanon
Qiang Ni, United Kingdom
Mahmood Niazi , Saudi Arabia
Vincent O. Nyangaresi, Kenya
Lu Ou , China
Hyun-A Park, Republic of Korea
A. Peinado , Spain
Gerardo Pelosi , Italy
Gregorio Martinez Perez , Spain
Pedro Peris-Lopez , Spain
Carla Ràfols, Germany
Francesco Regazzoni, Switzerland
Abdalhossein Rezai , Iran
Helena Rifà-Pous , Spain
Arun Kumar Sangaiah, India
Nadeem Sarwar, Pakistan
Neetesh Saxena, United Kingdom
Savio Sciancalepore , The Netherlands

De Rosal Ignatius Moses Setiadi ,
Indonesia
Wenbo Shi, China
Ghanshyam Singh , South Africa
Vasco Soares, Portugal
Salvatore Sorce , Italy
Abdulhamit Subasi, Saudi Arabia
Zhiyuan Tan , United Kingdom
Keke Tang , China
Je Sen Teh , Australia
Bohui Wang, China
Guojun Wang, China
Jinwei Wang , China
Qichun Wang , China
Hu Xiong , China
Chang Xu , China
Xuehu Yan , China
Anjia Yang , China
Jiachen Yang , China
Yu Yao , China
Yinghui Ye, China
Kuo-Hui Yeh , Taiwan
Yong Yu , China
Xiaohui Yuan , USA
Sherali Zeadally, USA
Leo Y. Zhang, Australia
Tao Zhang, China
Youwen Zhu , China
Zhengyu Zhu , China


Contents

Message Authentication and Network Anomalies Detection in Vehicular Ad Hoc Networks

Leonid Legashev , Irina Bolodurina , Lubov Zabrodina , Yuri Ushakov , Alexander Shukhman ,
Denis Parfenov , Yong Zhou, and Yan Xu 


Research Article (18 pages), Article ID 9440886, Volume 2022 (2022)

Research on Medical Waste Supervision Model and Implementation Method Based on Blockchain

Hui Wang, Longshuai Zheng, Qihong Xue, and Xueqing Li 






Research Article (16 pages), Article ID 5630960, Volume 2022 (2022)

Lightweight and Anonymous Mutual Authentication Protocol for Edge IoT Nodes with Physical Unclonable Function

Hongyuan Wang, Jin Meng, Xilong Du, Tengfei Cao, and Yong Xie 

Research Article (11 pages), Article ID 1203691, Volume 2022 (2022)

A Virtual Machine Migration Strategy Based on the Relevance of Services against Side-Channel Attacks

Ji-Ming Chen , Shi Chen , Xiang Wang , Lin Lin , and Li Wang 

Research Article (17 pages), Article ID 2729949, Volume 2021 (2021)

Combinatorial Spectrum E-Auction for 5G Heterogeneous Networks: A Zether-Based Approach

Zijun Zhao , Zuobin Ying, Zhiming Cai , and Jianfeng Ma


Research Article (10 pages), Article ID 1360560, Volume 2021 (2021)

Dominant Feature Selection and Machine Learning-Based Hybrid Approach to Analyze Android Ransomware

Tanya Gera , Jaiteg Singh , Abolfazl Mehbodniya , Julian L. Webber , Mohammad Shabaz , and
Deepak Thakur 



Research Article (22 pages), Article ID 7035233, Volume 2021 (2021)

Preventing Scan-Based Side-Channel Attacks by Scan Obfuscating with a Configurable Shift Register

Weizheng Wang , Yin Chen, Shuo Cai, and Yan Peng

Research Article (9 pages), Article ID 5222670, Volume 2021 (2021)

Blockchain-Enabled Intelligent Video Caching and Transcoding in Clustered MEC Networks

Yan Li  and Zheng Wan 


Research Article (17 pages), Article ID 7443260, Volume 2021 (2021)

Towards Trustworthy IoT: A Blockchain-Edge Computing Hybrid System with Proof-of-Contribution Mechanism

Huan Dai , Pengzhan Shi , He Huang, Ruyu Chen , and Jun Zhao 

Research Article (13 pages), Article ID 3050953, Volume 2021 (2021)

Security Analysis of Intelligent System Based on Edge Computing


Yibo Han, Weiwei Zhang, and Zheng Zhang 

Research Article (10 pages), Article ID 1224333, Volume 2021 (2021)


SSGD: A Safe and Efficient Method of Gradient Descent

Jinhuan Duan, Xianxian Li , Shiqi Gao, Zili Zhong, and Jinyan Wang 
Research Article (11 pages), Article ID 5404061, Volume 2021 (2021)


A Lightweight and Secure Anonymous User Authentication Protocol for Wireless Body Area Networks

Junsong Zhang , Qikun Zhang, Zhigang Li, Xianling Lu , and Yong Gan
Research Article (11 pages), Article ID 4939589, Volume 2021 (2021)

Blockchain-Based Key Management and Green Routing Scheme for Vehicular Named Data Networking

Hao Liu, Rongbo Zhu , Jun Wang, and Wengang Xu
Research Article (13 pages), Article ID 3717702, Volume 2021 (2021)

Classification of Abnormal Traffic in Smart Grids Based on GACNN and Data Statistical Analysis

F. F. Hu, S. T. Zhang, X. B. Lin, L. Wu, and N. D. Liao 
Research Article (19 pages), Article ID 9927325, Volume 2021 (2021)

Early Rumor Detection Based on Deep Recurrent Q-Learning

Wei Wang , Yuchen Qiu, Shichang Xuan , and Wu Yang 
Research Article (13 pages), Article ID 5569064, Volume 2021 (2021)

STQ-SCS: An Efficient and Secure Scheme for Fine-Grained Spatial-Temporal Top- k Query in Fog-Based Mobile Sensor-Cloud Systems

Jie Min , Junbin Liang , Xingpo Ma , and Hongling Chen 
Research Article (16 pages), Article ID 9939796, Volume 2021 (2021)




A Detection Approach for Vulnerability Exploiter Based on the Features of the Exploiter

Jinchang Hu , Jinfu Chen , Sher Ali , Bo Liu , Jingyi Chen , Chi Zhang , and Jian Yang 
Research Article (14 pages), Article ID 5581274, Volume 2021 (2021)

Blockchain-Based Efficient Device Authentication Protocol for Medical Cyber-Physical Systems

Fulong Chen , Yuqing Tang , Xu Cheng , Dong Xie , Taochun Wang , and Chuanxin Zhao 
Research Article (13 pages), Article ID 5580939, Volume 2021 (2021)

ECLB: Edge-Computing-Based Lightweight Blockchain Framework for Mobile Systems

Qingqing Xie , Fan Dong , and Xia Feng 
Research Article (15 pages), Article ID 5510586, Volume 2021 (2021)

Research Article

Message Authentication and Network Anomalies Detection in Vehicular Ad Hoc Networks

Leonid Legashev ¹, Irina Bolodurina ¹, Lubov Zabrodina ¹, Yuri Ushakov ¹,
Alexander Shukhman ¹, Denis Parfenov ¹, Yong Zhou,² and Yan Xu ²

¹Faculty of Mathematics and Information Technologies, Orenburg State University, Orenburg 460018, Russia

²School of Computer Science and Technology, Anhui University, Hefei 230601, China

Correspondence should be addressed to Leonid Legashev; silentgir@gmail.com

Received 3 June 2021; Revised 13 January 2022; Accepted 15 January 2022; Published 24 February 2022

Academic Editor: Mamoun Alazab

Copyright © 2022 Leonid Legashev et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Intelligent transport systems are the future in matters of safe roads and comfortable driving. Integration of vehicles into a unified intelligent network leads to all kinds of security issues and cyber threats common to conventional networks. Rapid development of mobile ad hoc networks and machine learning methods allows us to ensure security of intelligent transport systems. In this paper, we design an authentication scheme that can be used to ensure message integrity and preserve conditional privacy for the vehicle user. The proposed authentication scheme is designed with lightweight cryptography methods, so that it only brings little computational and communication overhead. We also conduct experiments on vehicular ad hoc network segment traffic generation in OMNeT++ tool and apply up-to-date machine learning methods to detect malicious behavior in a given simulated environment. The results of the study show high accuracy in distributed denial-of-service attack detection.

1. Introduction

The rapid development in the field of mobile devices, sensors, and 5G networks [1] allows incorporating computational nodes into wireless ad hoc network. A network without preexisting infrastructure is called a mobile ad hoc network (MANET); it consists of mobile devices capable of establishing connections between arbitrary nodes. Ad Hoc On-Demand Distance Vector (AODV), Destination-Sequenced Distance-Vector Routing (DSDV), Optimized Link-State Routing (OLSR), and Dynamic Source Routing (DSR) protocols are used for routing at the network layer in MANET. One of the important areas in MANET is vehicular ad hoc networks (VANETs), which represent intelligent transport system where each vehicle is considered as a mobile node. Potential VANET applications include road condition warnings, collision alerts, accident alerts, road congestions, driver assistance systems, and infotainment systems. Each vehicle in VANET is equipped with a set of sensors and constantly exchanges crucial information with

other nodes all over the network. These nodes may include fixed roadside units (RSUs), base station units (BSUs), trusted authority (TA) or control center (CC), and drones as mobile BSUs [2, 3]. It is very important to pay attention to security issues in VANET because the consequences of a network attack on the road can be unfortunate.

Complex research on VANET security issues may be divided into two directions. The first one is related to assistance of vehicle communication and vehicle privacy based on intelligent anonymous authentication and key agreement for 5G/beyond 5G (B5G) vehicular ad hoc networks. The second direction is related to machine learning algorithm applications in network threat detection and classification.

Modern trends in the creation of network connectivity of an increasing number of devices and the rise of the Internet of Things (IoT) and the Internet of Everything (IoE) required the development of new approaches to organizing network interaction. In cases where the network can contain several thousand devices, many of which are also intermediate for traffic transmission, traditional approaches can

be inefficient and slow. In the case of unstable links and nonstationary nodes, traditional mobile networking methods such as BATMAN-adv, OLSR, and AODV can cause losses, delays, rings, and instability of the entire network. Traditional software-defined networks (SDNs) can also be unstable under these conditions, especially in reactive mode. To implement intelligent transmission of information over such networks, for example, in the form of delayed packet transmission, a combination of controlled and autonomous approaches is required. Vehicular distributed software-defined networks (VSDNs) are a combination of proactive SDN management for consistent precalculated routes, while local reactive mode is used in conjunction with neighbor detection methods through legacy protocols. General scheme of VSDNs in VANET segment is presented in Figure 1.

SDNs are mainly used in VSDNs in stable parts of the network and in virtualization infrastructure, especially for Network Function Virtualization (NFV) modules and edge computing. When an NFV module is used as an edge virtual machine or a container running on network equipment, the requirements for the selection and routing of network flows passing through this module can be implemented only by SDN infrastructures managed by the controller. Since traffic can pass through the balancing nodes and be routed to the endpoint through various communication channels, it is important to have complete information about all network flows to a specific destination. When using distributed networks, they can contain several controllers with state synchronization (for example, via KV-storages); in this case, separate synchronization of applications related to packet and flow analysis is required to intercept the maximum possible number of directions of flow vectors. At the same time, since the traffic volumes of modern applications can exceed the capabilities of their analysis in real time, selective or only header preliminary analysis of packets and consolidation of data from all distributed controllers into a single storage is required, which will be used by many streaming analyzers.

2. Related Work

In order to ensure secure communication between intelligently connected vehicles, a public key cryptography (PKC) mechanism was proposed. In [4], the traditional PKC was proposed to implement self-certified public key cryptography (SCPCK) for online registration of multiserver architecture and to ensure the security of various mobile service applications.

The traditional PKC mechanism can realize secure communication; however, the mechanism suffers from various drawbacks caused by managing a large number of user certificates. In [5], Shamir put forward the concept of ID-based PKC. In [6, 7], a bilinear pairing ID-based PKC mechanism was proposed to achieve the required privacy for vehicles. In [8], an improved scheme ID-based PKC mechanism without bilinear pairing was proposed. The improved scheme is not required to utilize bilinear pairing operations without lack of security and privacy protection,

and the total computational cost of signature and authentication is constant for single message and n messages. In [9], an improved message authentication scheme together with a system secret key updating scheme was proposed to optimize the performance and security of V2V authentication process. Although the ID-based PKC avoids the problem of managing certificates brought by the traditional PKC mechanism, it still brings the problem of key escrow.

In 2013, Al-Riyami [10] first proposed the concept of certificateless PKC mechanism, which avoids the certificate management problems brought by traditional PKC mechanism and the key escrow problems caused by ID-based PKC mechanism. In [11], an anonymous authentication scheme based on certificateless PKC mechanism was proposed by using bilinear pairing operations. In [12, 13], all implement batch authentication without bilinear pairing based on certificateless PKC mechanism were proposed. In [14], a new authentication scheme without bilinear pairings was proposed. In [15], a reliable and efficient secure content sharing scheme for 5G-enabled VANETs was proposed. In [16], a lightweight and secure authenticated key agreement scheme for securing V2V and V2I communications simultaneously was proposed.

In order to ensure the efficiency and safety of VANETs, there are a large number of schemes for batch authentication of messages. In [17], Zhang et al. proposed a distributed aggregate batch authentication scheme. By dividing the received message into multiple subsets and then aggregating multiple subsets for batch authentication. In [18], an efficient batch authentication scheme based on elliptic curve cryptography was proposed. A proxy-based batch authentication scheme was proposed in [19], where some vehicles were selected as the proxy vehicles, whose message signatures were then verified by roadside units in batches.

Recently, there is a lot of research dedicated to machine learning methods' application in network threat detection. Montenegro J. et al. [20] applied machine learning techniques and trust model metrics to detect fake position attacks in VANETs. The same problem was solved by Singh P. K. et al. [21] using machine learning techniques on VeReMi dataset to detect false position information broadcast to the other vehicles. A Ghaleb F. et al. [22] used the random forest algorithm to train intrusion detection system classifiers on each vehicle node with the overall goal of reducing the communication overhead. Nandy T. et al. [23] also proposed a trust-based collaborative intrusion detection system with k-nearest neighbors nonlinear classifier to identify intruders in real time. To detect various malevolent attacks, Sharma S. et al. [24] proposed a Multicluster Head anomaly based intrusion detection system with Dolphin Swarm Algorithm optimization technique. Zhang T. et al. [25] in their research proposed a privacy-preserving machine learning based collaborative intrusion detection system for VANETs. Zhang D. et al. [26] proposed a software-defined trust-based deep reinforcement learning framework for VANET issues related to performance degradation. Belenko V. et al. [27] proposed approach to generate VANET dataset with various scenarios of cyber attacks for the ns-3 network simulator. Singh P. K. et al. [28]

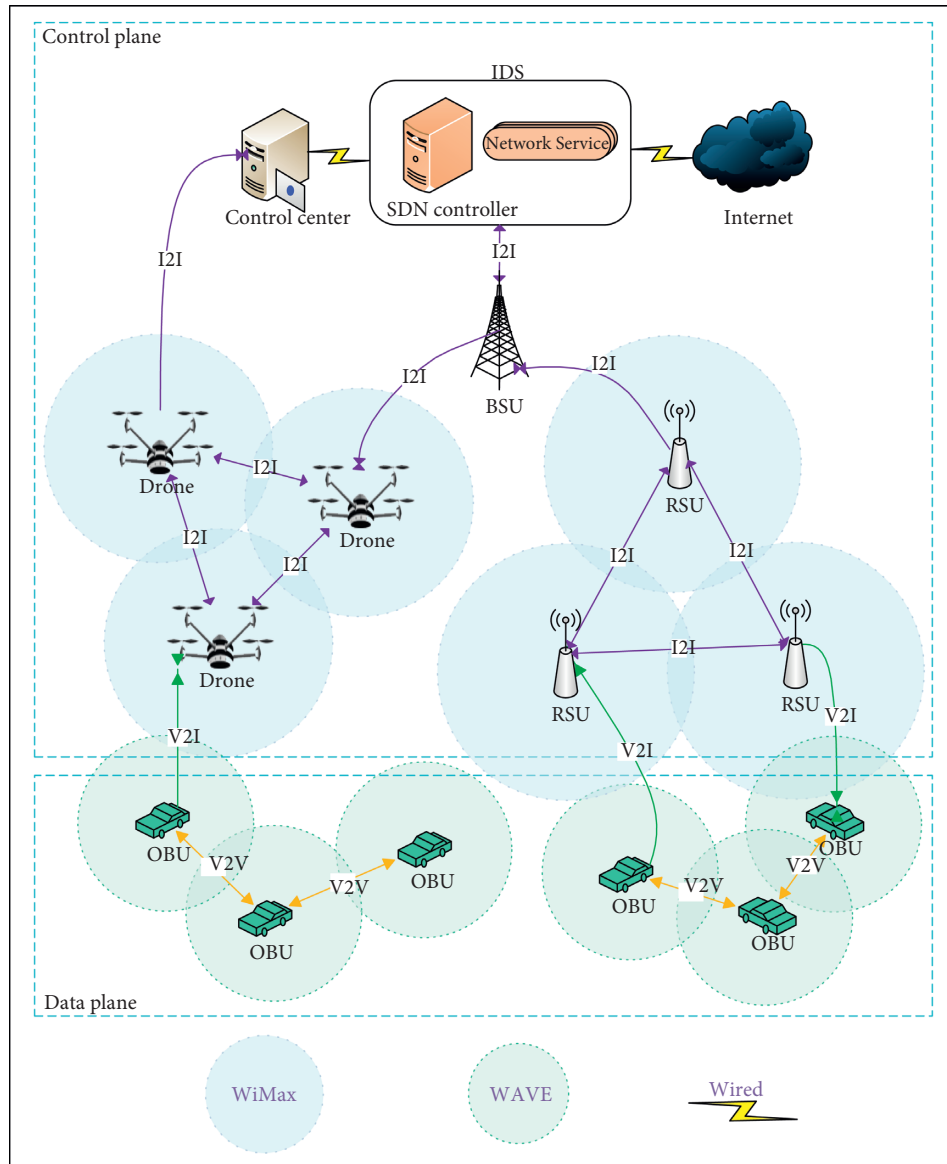


FIGURE 1: Scheme of VDSDNs in VANET segment.

also generated scenario of multihop communication on ns-3 network simulator to detect wormhole attacks in VANET using KNN and support vector machine models. Kumar S. et al. [29] presented a vehicular jamming system model with usage of CatBoost machine learning algorithm to predict the coordinates of jamming vehicle. Rehman A. et al. [30] described a novel approach to detect intrusion attacks on the CAN bus using convolutional neural network and attention-based gated recurrent unit. Jhaveri R. et al. [31] proposed a bandwidth contract-based framework to provide resilience to violation of the bandwidth requirements of the traffic flows in vehicular ad hoc networks.

Different types of simulation tools can be used to generate reliable VANET traffic and experiment over many types of scenarios within intelligent transport system. Akhtar et al. [32] presented simulation model of microscopic mobility VANET segment by using SUMO [33] traffic simulation package and Freeway Performance Measurement

System database. Michaeler et al. [34] presented 3-dimensional driving simulator based on Open-StreetMap data, which integrates VANET communication capabilities. Buse et al. [35] proposed event-driven simulator for the advanced driver assistance system development. To ensure reliable driver assistance systems, Obermaier et al. [36] presented an approach for testing VANET devices and the applications in hardware in the Loop environment using OMNeT++ simulation tool [37] and the VANET model Artery. Fahad et al. [38] proposed a new scheme based on compressed fuzzy logic method to enhance AODV routing decisions in VANET. Maratha et al. [39] conducted performance study of AODV, DSDV, and DSR MANET protocols using NCTUns 6.0 network simulator [40]. Raj et al. [41] simulated various routing protocols using ns-3 simulator [42] and SUMO package and studied performance metrics such as Packet Delivery Ratio, Throughput, and End-to-End Delay.

Table 1 contains the coverage of machine learning methods' applications of network attack detection in some recent publications, including Naive Bayes (NB), logistic regression (LR), support vector machine (SVM), random forest (RF), CatBoost (CB), AdaBoost (AB), and gradient boosting (GB). As you can see, existing works are limited to the usage of individual classifiers only. In our current research, we will make a comparison of the most up-to-date classifiers and ensemble methods for the multiclass classification problem of distributed denial-of-service attack detection using simulated VANET environment.

To study security issues in vehicular ad hoc networks and detect DDoS attacks, we make the following contributions:

Authentication method: we proposed an anonymous authentication scheme based on elliptic curve encryption to meet the security requirements of vehicular ad hoc networks providing the least time cost on signing and verifying a message.

Simulation of VANET dataset: using OMNeT++ simulation tool we simulated segment of VANET and implemented three types of popular network attacks which degrade overall performance of intelligent transport system by flooding vehicles with great amount of generated messages.

Experimentation and evaluation: we recorded network flows information from the nodes in simulated segment of VANET and conducted experiments on multiclass classification using the most advanced classifiers and ensembles of classifiers.

The structure of the rest of the paper is as follows: Section 3 introduces the intelligent anonymous authentication scheme. In Section 4, we describe the proposed generation of VANET segment traffic in OMNeT++ simulation tool and present results of multiclass classification of DDoS attacks. Section 5 gives the conclusion.

3. Intelligent Authentication Methods for Vehicular Ad Hoc Networks

3.1. System Model. The considered system consists of three parts as shown in Figure 2, which can be divided into three layers. The first layer contains TA which communicates with EA and vehicles over a secure channel by wired connections. The second layer includes EA, which communicates with TA and vehicles by the secure channel. The third layer involves multiple vehicles which mainly communicate with cluster head (CH) by DSRC protocol. The definitions of the roles involved in the considered model are as follows.

TA: It is a trusted authority that consists of a Key Generation Center (KGC) and a Trace Authority (TRA) in the practical environment. The KGC initializes the system and generates all public parameters and private keys. The TRA extracts the real identities of malicious vehicles by pseudo identities in the case of controversial traffic events.

EA: It is an edge authority that contains multiple mobile devices (e.g., mobile phones, laptops, and other

electronic devices). The EA contains enough computing and storage sources which are used to handle reputation update and reveal vehicles real identities by pseudo identities in the disputed traffic environment.

Vehicles: We assume that multiple vehicles can form a cluster in the same area. In order to avoid repeated calculations of multiple vehicles in the cluster, the EA selects the vehicle with good network resources (e.g., high network bandwidth, constant speed, and suitable location) as the CH vehicle. The CH predownloads the road condition related data required by the vehicle; then, the remaining vehicles and the CH authentication reduce the redundancy calculation.

3.2. Our Proposed Authentication Scheme. In order to meet the security requirements of VANETs, we design an anonymous authentication scheme using an online certificateless signature technology based on prioritization. Our proposed scheme consists of several algorithms including setup, pseudo identity generation/partial key extraction, sign, batch verification, and revocation/update of revocation list. The scheme details can be shown as follows.

3.2.1. Setup. TA initializes the system, generates public parameters for the system, and then sends related parameters to EA by security channel. The details are as follows.

- (1) The TA chooses a cyclic addition group G_1 with order q generated by P . Let $E: y^2 = x^3 + ax + b \pmod{n}$, $a, b \in F_n$, be an elliptic curve over the finite field F_n , where n indicates a large prime number. All the points of E and an infinity point O are in the group G_1 .
- (2) The KGC selects two random numbers as x, y and calculates $PK_{TA} = yP$. The KGC generates public-private key pairs for EA: $sk_{EA} = s, pk_{EA} = sP$.
- (3) The TA selects two one-way hash functions: $h_1: \{0, 1\}^* \rightarrow \{0, 1\}^*, h_2: \{0, 1\}^* \rightarrow Z_q$.
- (4) The TA publishes $\{P, PK_{TA}, pk_{EA}, h_1, h_2\}$ as the system parameter and sends (x, sk_{EA}) to EA through the secure channel.

3.2.2. Pseudo Identity Generation and Partial Key Extraction. The EA communicates with vehicles online to generate pseudo identities for vehicles, and the TA generates a partial key for the vehicle through its KGC. The details are as follows:

- (1) The vehicle calculates partial public-private key pairs for itself: $sk_{i,1} = s_1, pk_{i,1} = s_1P$.
- (2) Vehicle i encrypts $(RID_i, pk_{i,1})$ with its own private key and EA's public key and sends $\{Enc_{sk_{i,1}, pk_{EA}}(RID_i, pk_{i,1}), TS_i\}$ to the EA, where $Enc(\cdot)$ is the asymmetric encryption method, RID_i is the real identity of vehicle i , and TS_i represents the current timestamp.

TABLE 1: Machine learning classifiers' application in network attack detection.

	So et al. [43]	Grover et al. [44]	Zeng et al. [45]	Singh et al. [21]	Montenegro et al. [20]	Sharshembiev et al. [46]	Tama et al. [47]	The proposed scheme
IBK		+						
J-48		+					+	
NB		+						
LR				+		+		
SVM	+		+	+				
KNN	+				+	+	+	+
RF		+					+	+
CB								+
XGB							+	+
LGBM								+
AB		+						+
GB							+	+

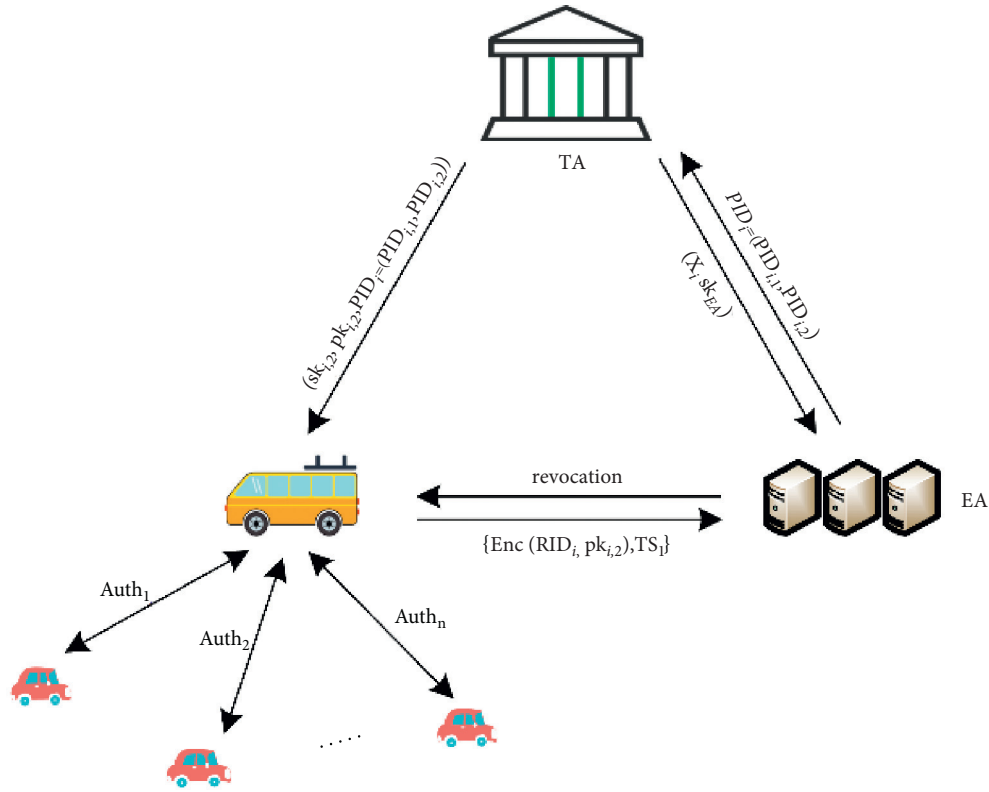


FIGURE 2: The system model.

- (3) The EA checks the timestamp for freshness by determining whether $|TS - TS| \leq \Delta T_i$ holds or not. If not, the EA stops. Otherwise, it utilizes its own private key and vehicle i 's public key to decrypt $Dec_{sk_{i,1}, pk_{EA}}(RID_i, pk_{i,1})$, calculates the pseudo identity $PID_{i,1} = xP$, $PID_{i,2} = RID_i \oplus h_1(xpk_{i,1})$, and then sends $PID_i = (PID_{i,1}, PID_{i,2})$ to the TA.
- (4) The TA selects a random number s_2 and calculates partial public key $pk_{i,2} = s_2P$ and partial private key $sk_{i,2} = s_2 + h_2(PID_i, pk_{i,2})y$, and then the TA sends partial public-private key pairs and pseudo identities $\{pk_i = (pk_{i,1}, pk_{i,2}), sk_i = (sk_{i,1}, sk_{i,2}), PID_i\}$ to the vehicle.

3.2.3. *Sign.* Each vehicle calculates the signature using the following steps:

- (1) Vehicle i calculates the signature:

$$S_i = sk_{i,2}h_1(PID_i \| m_i \| TS_i) + sk_{i,1}, \quad (1)$$

$$RV_i = (S_i ID_i \| m_i \| TS_i) \oplus h_1(RU_j),$$

where RU_j is used to check whether vehicle i is in the revocation list.

- (2) Vehicle i sends $\{S_i, PID_i, m_i, TS_i, RV_i, h_1(RU_i)\}$ to the CH.

3.2.4. *Batch Verification.* The CH batch verifies the vehicles in the cluster using three steps:

- (1) It checks whether vehicle i is in the revocation list:

$$RV_i \oplus h_1(RU_j) == m_i \| S_i \| PID_i \| TS_i. \quad (2)$$

If the equation is true, which indicates that vehicle i is not in the revocation list, CH continues to execute the next step; otherwise, it stops making the remaining steps.

$$S_i P = (pk_{i,2} + h_1(PID_i \| m_i \| TS_i) PK_{TA}) \cdot h_1(PID_i \| m_i \| TS_i) + pk_{i,1}. \quad (3)$$

- (3) The CH performs the following check for every vehicle i , $i \in [1, n]$:

$$\sum_{i=1}^n S_i P == \sum_{i=1}^n (pk_{i,2} + h_1(PID_i \| m_i \| TS_i) PK_{TA}) \cdot h_1(PID_i \| m_i \| TS_i) + \sum_{i=1}^n pk_{i,1}. \quad (4)$$

If the equation is true, it means that the vehicle is verified, and the relevant data can be obtained from the CH; otherwise, the next step is continued.

3.2.5. *Revocation/Update of Revocation List.* If the CH verification equation does not hold, we need to trace the specific vehicle and update the revocation list. We assume the vehicle i is a malicious vehicle, then revoke it, and update revocation, which includes three steps:

- (1) The CH sends (PID_i, pk_i) to the EA; then, the EA extracts the real identity of the vehicle i by calculating $RID_i = PID_{i,2} \oplus h_1(xpk_{i,1})$ and updating the revocation list.
- (2) If the EA has a single point of failure, the CH sends (PID_i, pk_i) to the TA. The TA calculates $RID_i = PID_{i,2} \oplus h_1(xpk_{i,1})$, then the real identity is sent to the EA, and the revocation list is updated.
- (3) If the vehicle is not on the revocation list, the EA calculates $H_u = h_1(RU_j) \oplus h_1(xpk_{i,1})$ for the vehicle. After the vehicle gets H_u , $h_1(RU_j)$ is obtained by calculating $h_1(RU_j) = H_u \oplus h_1(sk_{i,1}, PID_{i,1})$.

3.3. *Security Analysis.* In this subsection, we discuss the security requirements for our proposed scheme.

- (1) Message authentication: In our proposed scheme, the vehicles are authenticated by the CH and thus obtain the relevant data. Multiple vehicles send messages to be verified to the CH with $\{S_i, PID_i, m_i, TS_i, RV_i, h_1(RU_i)\}$. Using the batch verification method, the message receiver can verify the legality of the message.

- (2) It checks if the signature is valid or not by checking the equation $|TS - TS_i| \leq \Delta T$. If so, the timestamp is fresh, and the CH continues to execute the next step; otherwise, it stops making the remaining steps.

- (3) Batch verification: the batch verification method is as follows:

- (1) The CH calculates $h_1(PID_i \| m_i \| TS_i)$ for the vehicle i .
- (2) The CH verification for the vehicle i is as follows:

- (2) Conditional privacy: Our scheme achieves conditional privacy protection. If a malicious vehicle i appears during verification, the CH can send (PID_i, pk_i) to the EA; then, the EA extracts the real identity of the vehicle i by calculating $RID_i = PID_{i,2} \oplus h_1(xpk_{i,1})$. The proposed scheme achieves double trace. When the EA has a single point of failure, the CH sends (PID_i, pk_i) to the TA. The TA calculates $RID_i = PID_{i,2} \oplus h_1(xpk_{i,1})$.

- (3) Identity privacy preserving: In the proposed scheme, CM communicates with CH using pseudo identities. The EA calculates the pseudo identities for each vehicle: $PID_{i,1} = xP$, $PID_{i,2} = RID_i \oplus h_1(xpk_{i,1})$. Since solving *CDHP* is difficult, it is not feasible for any vehicle to extract the real identity of another vehicle via pseudo identity, except for the trusted authority.

- (4) Strong privacy preserving: Our proposed scheme achieves strong privacy preserving. Since the EA only knows the secret value x and does not know y , even if the EA is compromised, the adversary cannot obtain the privacy information of any vehicle.

3.4. *Performance Analysis.* In this subsection, we evaluate the computational overhead and communication load of the proposed scheme and compare it with three related schemes [17, 18, 48]. In [48], a distributed aggregation privacy protection authentication scheme (DAPPA) is proposed. All the signatures are divided into multiple subsets, and the aggregated signature is verified. In [17], an efficient certificateless batch authentication scheme without pairing (ECLA), which utilizes the elliptic curve cryptography to achieve batch authentication, is proposed. In [18], a new

identity-based message authentication scheme (ID-MAP) is proposed; it uses the message authentication of proxy vehicles to significantly reduce the computational cost of roadside units. The specific comparison process is described in the next two sections.

In this section, we analyze the computational overhead of the proposed scheme and compare it with [17, 18, 48]. We choose a bilinear pairing, $e: G_1 \times G_1 \rightarrow G_2$, to achieve the security level of 80 bits, where G_1 is an additive group generated by a point p^* with the order q^* on the super singular elliptic curve $E: y^2 = x^3 + ax + b \pmod{p^*}$, p^* is a 512-bit prime number, and q^* is a 160-bit Solinas prime number. Meanwhile, we choose a nonsingular elliptic curve $E: y^2 = x^3 + ax + b \pmod{p}$, $a, b \in F_p$, where all points on the elliptic curve (including an infinity point) are on an addition group G whose generator is P , and p, q are 160 bits to achieve security level of 80 bits. To simplify the expression, we predefine the following symbols:

T_{bp} : the time to perform a bilinear pairing operation.

T_{bp}^{sm} : the time to perform a scale multiplication operation related to pairing-based cryptography (PBC).

T_{bp}^{pa} : the time to perform a point addition operation of the bilinear pairing.

T_{mtp} : the time to perform a map-to-point hash function operation related to PBC.

T_{ecc}^{sm} : the time to perform a scale multiplication operation related to ECC.

T_{ecc}^{pa} : the time to perform a point addition operation related to ECC.

We can obtain these cryptographic operations' execution time using MIRACL library [19], with the platform of 3.4 GHZ i7-4770. The execution time of the above cryptographic operations is $T_{bp} = 4.211ms$, $T_{bp}^{sm} = 1.709ms$, $T_{bp}^{pa} = 0.007ms$, $T_{mtp} = 4.406ms$, $T_{ecc}^{sm} = 0.442ms$, $T_{ecc}^{pa} = 0.0018ms$. Next, we analyze the details of computational overhead for DAPPA, ECLA, ID-MAP, and the proposed scheme.

The DAPPA scheme was proposed in [48] based on bilinear pairing. In this scheme, signing a message requires the signer to perform five scale multiplication operations of the bilinear pairing, one point addition operation of the bilinear pairing, and two map-to-point hash function operations: $5T_{bp}^{sm} + T_{bp}^{pa} + 2T_{mtp} = 17.364ms$. When the verifier receives n messages, it first divides the n messages into n^* subsets; each subset includes n/n^* messages to be verified. To verify n messages, the verifier needs to perform two bilinear pairing operations, n scale multiplication operations of the bilinear pairing, $(n - n^*)$ point addition operations of the bilinear pairing, and $2n$ map-to-point hash function operations (in this case, we take $n^* = 1$): $2T_{bp} + nT_{bp}^{sm} + 2T_{bp}^{pa} + 2nT_{mtp} = (10.521n + 8.436)ms$.

The ECLA scheme was proposed in [17] based on elliptic curve encryption. Signing a message in this scenario requires the signer to perform two scale multiplication operations of the elliptic curve cryptography: $2T_{ecc}^{sm} = 0.884ms$. When the verifier receives n messages, the verification of n messages requires the verifier to perform $5n$ scale multiplication

operations of the elliptic curve cryptography and $3n$ point addition operations of the elliptic curve cryptography: $(5T_{ecc}^{sm} + 3T_{ecc}^{pa})n = (2.2154n)ms$.

The ID-MAP scheme was proposed in [18] based on elliptic curve encryption. In this scenario, the proxy vehicle needs to perform $(l + 6)$ scale multiplication operations of the ECC: $(l + 6)T_{ecc}^{sm} = 7T_{ecc}^{sm} = 3.094ms$, where l represents the number of proxy vehicles, and we set it as 1. When the roadside unit receives n messages, it needs to perform $5n/l$ scale multiplication operations of ECC to verify n messages: $5n/lT_{ecc}^{sm} = 5nT_{ecc}^{sm} = (2.21n)ms$. For the proposed OAAS based on elliptic curve encryption, to implement the signature of a message in this scheme, the CM should perform one scale multiplication operation of ECC: $T_{ecc}^{sm} + T_{ecc}^{pa} = 0.4438ms$. When the CH receives N messages to be verified, where there are n messages in the sequence with high priority, the CH verifies that n messages need to perform $2n$ scale multiplication operations of the elliptic curve cryptography and $(n + 1)$ point addition operations of ECC: $2nT_{ecc}^{sm} + (n + 1)T_{ecc}^{pa} = (0.8858n + 0.0018)ms$.

In order to more intuitively observe the computational performance of our proposed scheme, Figure 3 compares the total cost of the four schemes. As can be seen from Figure 3, our proposed scheme has the least time cost in signing and verifying a message.

Next, we analyze the computational complexity of the main steps of the proposed authentication scheme. Suppose that the security parameter is K . The most computationally expensive operations in the authentication scheme mainly include the scale multiplication operations of ECC and the multiplication operation of two big numbers, whose computational complexities are $O(\log_2 K)$ and $O(K)$, respectively. Other operations such as hash and XOR operations have $O(1)$ computational complexity. Therefore, the computational complexities of setup, pseudo identity generation and partial key extraction, sign, batch verification, and revocation/update of revocation list steps are $O(\log_2 K)$, $O(K)$, $O(K)$, $nO(\log_2 K)$, and $O(K)$, respectively, where n denotes the number of vehicles.

4. Intelligent Algorithm for DDoS Attack Detection in VANET Dataset

4.1. Simulation of VANET Segment in OMNeT++. Another problem of our research is to develop methods for detecting distributed denial-of-service attacks in software-defined vehicular ad hoc networks. To solve this problem we decided to generate VANET dataset which is suitable for our research purposes. VANET dataset generation includes the following steps:

- (1) Simulation of VANET segment in different scenarios.
- (2) Getting simulation results in the form of PCAP files.
- (3) PCAP files processing and traffic flow feature extraction.
- (4) Formation of single .csv dataset with the obtained features.

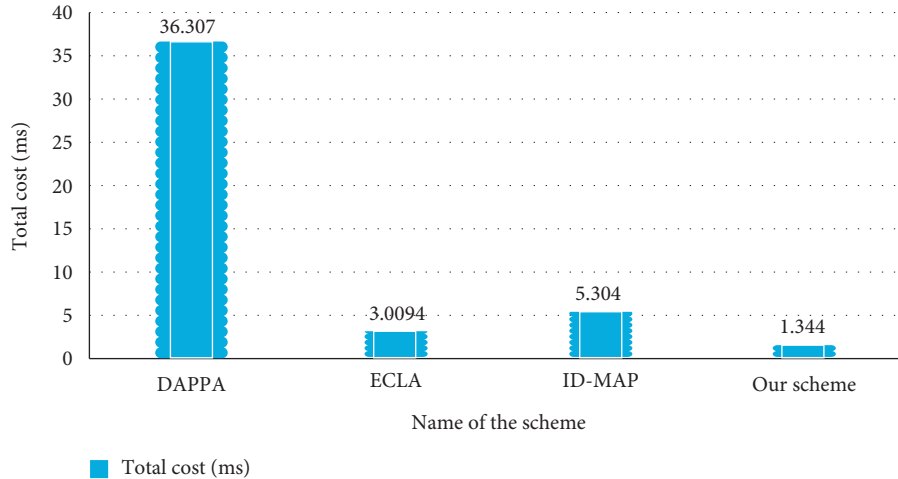


FIGURE 3: Total computation cost of the four schemes.

Typical solution of VANET simulation includes OMNeT++, INET framework [49], SUMO, and Veins framework [50]. As a first approximation, we will consider OMNeT++ and INET solution to build small segment of mobile ad hoc network with DoS and DDoS attack implementation scenarios. Our solution is based on MANET routing protocols, in particular AODV showcase.

Generally, simulation in OMNeT++ consists of three steps:

- (1) Set network elements in .ned file to describe network model.
- (2) Set general settings and each element's settings in omnetpp.ini file.
- (3) Perform simulation and record the results.

5. Perform Simulation and Record the Results

The general settings of our VANET test segment are presented in Table 2.

We will consider the following simulation case: we have an immobile vehicle (source node) which had an accident and is trying to send information to the base station unit (destination node) using other vehicles (up to ten) as relay nodes. According to this, the elements of our VANET test segment are presented in Table 3.

Ten vehicles are freely moving across the given square area in random directions. To simulate this, we need to set vehicle movement to linear mobility type (see Table 4). We can also change type of node mobility to the VehicleMobility; in this case, it is necessary to set waypointFile for each vehicle with pairs of coordinates of each movement route around the designed area.

Our goal is to obtain the PCAP file of traffic flow for each node type, so we need to apply the PCAP recorder settings provided in Table 5. To use tools for network traffic flow feature extraction, we need to record data of sending and receiving packets.

The number of PCAP recorders is set to 4 because we will record information from source node, destination node,

TABLE 2: General setting of VANET segment.

WLAN bitrate	24 Mbps
Transmitter power	1 mW
Area size	800 m × 800 m
Routing protocol	AODV
Link type	IEEE 802.11 wireless LAN
Simulation time	1000 s

TABLE 3: VANET segment elements and their types.

Element	Type
Source	ManetRouter
Destination	ManetRouter
Nodes 1-10	ManetRouter
radioMedium	Ieee80211ScalarRadioMedium
Visualizer	IntegratedMultiVisualizer
Configurator	Ipv4NetworkConfigurator
PcapRecorder	PcapRecorder

TABLE 4: Mobile node movement settings.

Type	Linear mobility
Initial movement heading	Uniform (0 deg, 360 deg)
Speed	Uniform (23 mps, 24 mps)

TABLE 5: Traffic flow PCAP recorder settings.

pcapLinkType	101 # raw IP
pcapFile	Results/***.pcap"
moduleNamePatterns	"ipv4"
dumpProtocols	"ipv4"
numPcapRecorders	4

malicious node (in scenario of DoS attack), and arbitrary relay vehicle (let us say node 5). Option pcapLinkType = 101 allows us to record raw IP information of corresponding node.

AODV protocol operates with three types of messages: RREQ, RREP, and RERR. Source node is sending request



FIGURE 4: Demonstration of AODV protocol operation in OMNeT++.

message RREQ into the network, and relay nodes forward this message further, causing the building of temporary routes to the destination node. When destination node receives request, it sends RREP message back to the source node using a built temporary route. In case when a destination node is unreachable, a RERR message is used to notify other nodes in the network segment. In Figure 4, you can see an example of successful route building between source node and destination node using six relay vehicles. “Ping108” and “ping108-reply” lines are used for visual indication of established bidirectional connection between two nodes.

To implement denial-of-service attack, we will add single MaliciousNode element with ManetRouter type and the settings in Table 6.

MaliciousNode performs simple DoS attack of PingApp type, constantly sending dozens of packets to the source node. MaliciousNode is immovable and is placed closer to the source node in the area. In case of DDoS attack implementation, we will set five moving malicious nodes of ManetRouter type which inherit other vehicles movement settings mentioned earlier in the paper.

To prove the efficiency of implemented network attacks, we performed simulation in three scenarios (without attacks, with DoS attack, and with DDoS attack) and calculated the number of received pings (successful establishment of connection between source node and destination node) and the number of lost pings. Experiment results are presented in Table 7.

You can see that implementation of DoS attack with single malicious node interfered with network routing, and

TABLE 6: Malicious node settings.

Attack type	PingApp
destAddr	“Source”
Start time	5 s
Sending interval	0.01 s
Radio transmitter power	200 mW
Packet size	0.5 Mb

some connections were lost. Implementation of DDoS attack with five moving malicious nodes completely “paralyzed” the network segment, and no connections were established.

5.1. Traffic Flows Feature Extraction. After obtaining PCAP files with captured traffic flows, we need to extract the set of features which will be used in machine learning methods. Any unusual behavior in traffic flow patterns and rapidly increased/decreased values of features can be qualified as possible network threat. Each PCAP file contains basic features such as source IP and number of port, destination IP and number of port, number of used protocols, and flow duration in seconds. Other features (such as number of forward/backward packets, average length of forward/backward packets, and average length of forward/backward packets headers) can be calculated manually or obtained using some software. CICFlowMeter [51] is a powerful tool to extract up to 84 network traffic features from PCAP files. The final segment of the network includes 15 relay vehicles, 2 RSUs, 1 BSU, and 6 malicious nodes with three types of

TABLE 7: AODVroute building results.

Route reply message	Scenario 1, without attacks	Scenario 2, DoS attack	Scenario 3, DDoS attack
Received pings replies	18	8 (partial connection)	No connection
Pings lost	2	1 (partial connection)	No connection

TABLE 8: Malicious nodes' behavior in scenario 4.

No.	App	Target node	Operation time interval	Packet size interval	Sending frequency	Mobility option
1	PingApp	Source	200–400 s	40–50 KB	0.01 s	Fixed
2	PingApp	Destination	400–700 s	50–60 KB	0.01 s	Fixed
3	PingApp	RSU1	250–450 s	10–30 KB	0.001 s	Linear
4	PingApp	RSU2	850–1450 s	20–40 KB	0.001 s	Linear
5	UDPBasicApp	Source	2600–3650 s	30–50 KB	0.01–0.05 s	Fixed
6	UDPBasicApp	Destination	3000–3850 s	40–60 KB	0.01–0.03 s	Fixed

- (1) Calculate correlation matrix of 58 features.
- (2) if ($\text{corr_value} > 0.05$):
- (3) Select best features in correlation with “Label” column.
- (4) Build scatter plot matrix of the best selected features.
- (5) Carry out data preparation (data balancing).
- (6) Perform multiclass classification by separate classifiers.
- (7) Perform multiclass ensemble classification:
- (8) Bagging Classifier usage.
- (9) Voting Classifier usage.
- (10) Stacking Classifier usage.

ALGORITHM 1: Multiclass classification of DDoS attacks.

behavior. Settings of each malicious node are shown in Table 8. Values in each given interval are distributed uniformly. To obtain more data, we increased simulation time to 5000 s and performed scenario 4 with three types of network attacks.

Recorded PCAP files were processed with CIC-FlowMeter; in general, 58 features of traffic flows were calculated in the form of .csv dataset. Each processed dataset contains “Label” column which is filled manually depending on the node type. Regular traffic flows were labeled “Benign,” and malicious traffic flows were given three labels: “DDoS,” “Intensive DDoS,” and “UDP Flood” according to the type of network attack behavior. All processed files were concatenated into single dataset.

5.2. DDoS Attack Detection Using Machine Learning Methods. The final dataset of VANET segment contains 11212 rows: 8720 records of normal traffic, 965 records of DDoS attack, 820 records of Intensive DDoS attack, and 707 records of UDP Flood attack. In the first step, we dropped the following columns: “Src Port,” “Dst Port,” and “Protocol,” which were used in traffic flows labeling. We also dropped the columns with constant values. Since our traffic flows are categorical features, we performed label encoding from [“Benign”, “DDoS”, “Intensive DDoS”, “UDP Flood”] to [0, 1, 2, 3]. General approach to multiclass classification of network attacks includes the steps presented in Algorithm 1.

In the first three steps, we performed correlation analysis of our generated VANET dataset. The correlation matrix is shown in Figure 5. The seven best features were selected according to the set threshold corr_value (see Table 9 for details). “Active Max” feature with highest correlation is corresponding to the maximum value of time when traffic flow was active before becoming idle during simulation.

In step 4, we built plot pairwise relationships (scatter plot matrix) of the best selected features as is shown in Figure 6. It can be concluded that the distribution of the values of the best selected features by classes is visually distinguishable, and further multiclass classification is justified.

In step 5, we formed two datasets based on the original one: imbalanced dataset, Data; balanced dataset, DataSMOTE. To obtain a balanced dataset, the algorithm SMOTE [52] (Synthetic Minority Oversampling Technique) was used to synthesize new examples for the minority class (class with three types of network attacks) without duplication of data. The resulting dataset DataSMOTE contains 7831 records of normal traffic, 8174 records of DDoS attack, 8167 records of Intensive DDoS attack, and 8232 records of UDP Flood attack. Dataset Data remained intact. For both Data and DataSMOTE datasets, we performed a split into training and test subsets with default proportion of 3:1.

In step 6, we performed multiclass classification with 7 well-known classifiers: KNeighborsClassifier, RandomForestClassifier, CatBoostClassifier, XGBClassifier, LGBMClassifier, AdaBoostClassifier, and

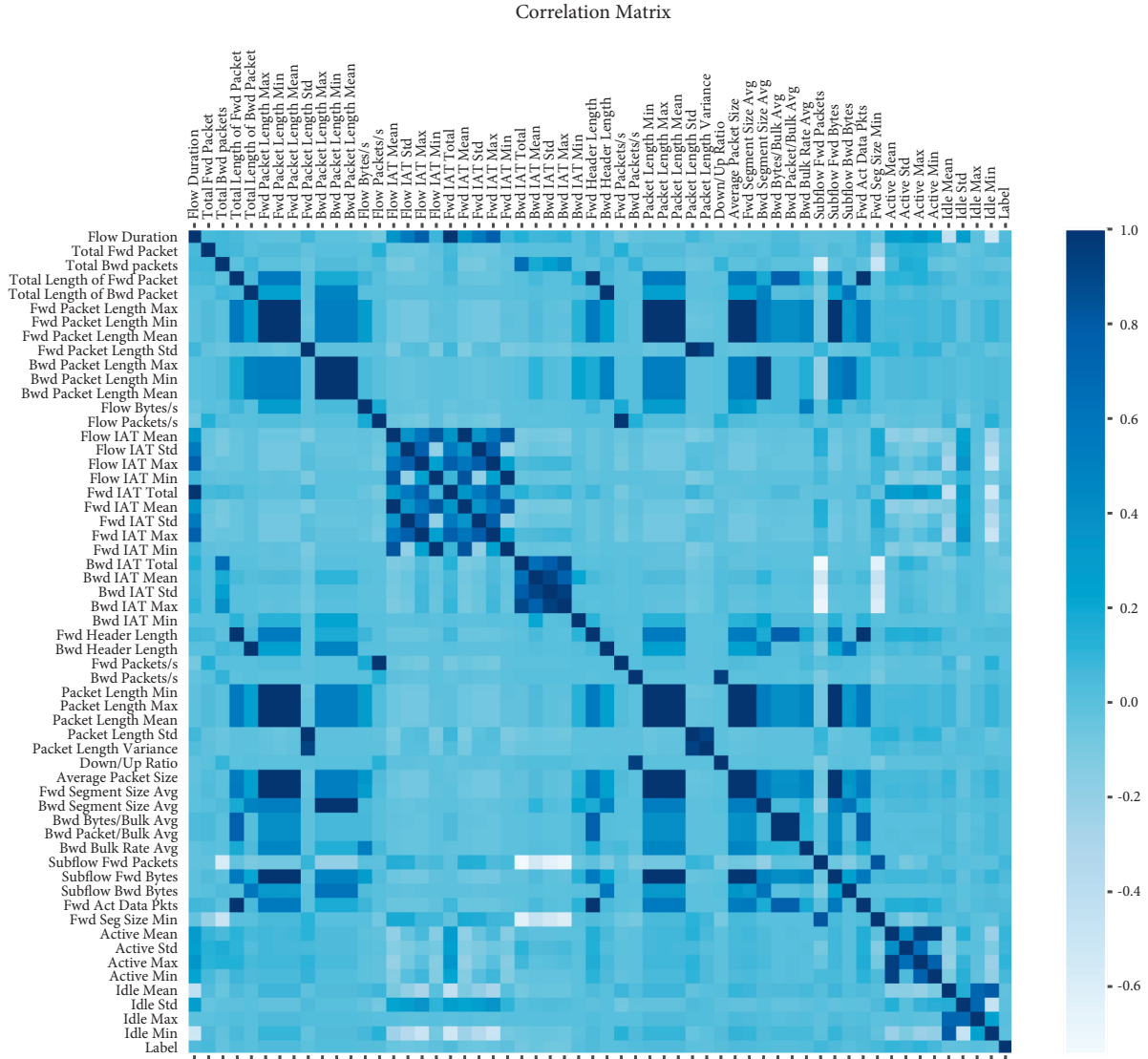


FIGURE 5: Correlation matrix of generated VANET dataset.

GradientBoostingClassifier. KNeighborsClassifier is implementation of k-nearest neighbors algorithm used for classification and regression problems. RandomForestClassifier is implementation of random forest meta-estimator algorithm. CatBoostClassifier is implementation of gradient boosting on decision trees algorithm developed by Yandex Company. XGBClassifier is implementation of parallel tree boosting algorithm. LGBMClassifier is implementation of gradient boosting model developed by Microsoft Company. AdaBoostClassifier is implementation of AdaBoost-SAMME algorithm. GradientBoostingClassifier is implementation of decision trees algorithms. Bentéjac C. et al. [53] performed comparative analysis of the family of gradient boosting algorithms in terms of speed and accuracy metrics. They concluded that LightGBM is the fastest classifier, but CatBoost shows the best results in generalization accuracy.

For each machine learning method, the optimal parameters were selected using the function GridSearchCV of the Python module sklearn. We built confusion matrices and

TABLE 9: Best selected features.

Feature	Correlation value
Total Fwd Packets	0.060131
Subflow Fwd Packets	0.078293
Fwd Seg Size Min	0.060144
Active Mean	0.092884
Active Std	0.084663
Active Max	0.104269
Active Min	0.066607

calculated the following statistical metrics for all 7 classifiers: precision, recall, F1-score, and accuracy (see Figures 7 and 8). LightGBM classifier shows the best classification results for the balanced dataset DataSMOTE with accuracy of 0.9180; its confusion matrix is shown in Figure 9.

In steps 6–10, to improve the accuracy of network attack detection, we built ensembles of the best models using the bagging, voting, and stacking methods.

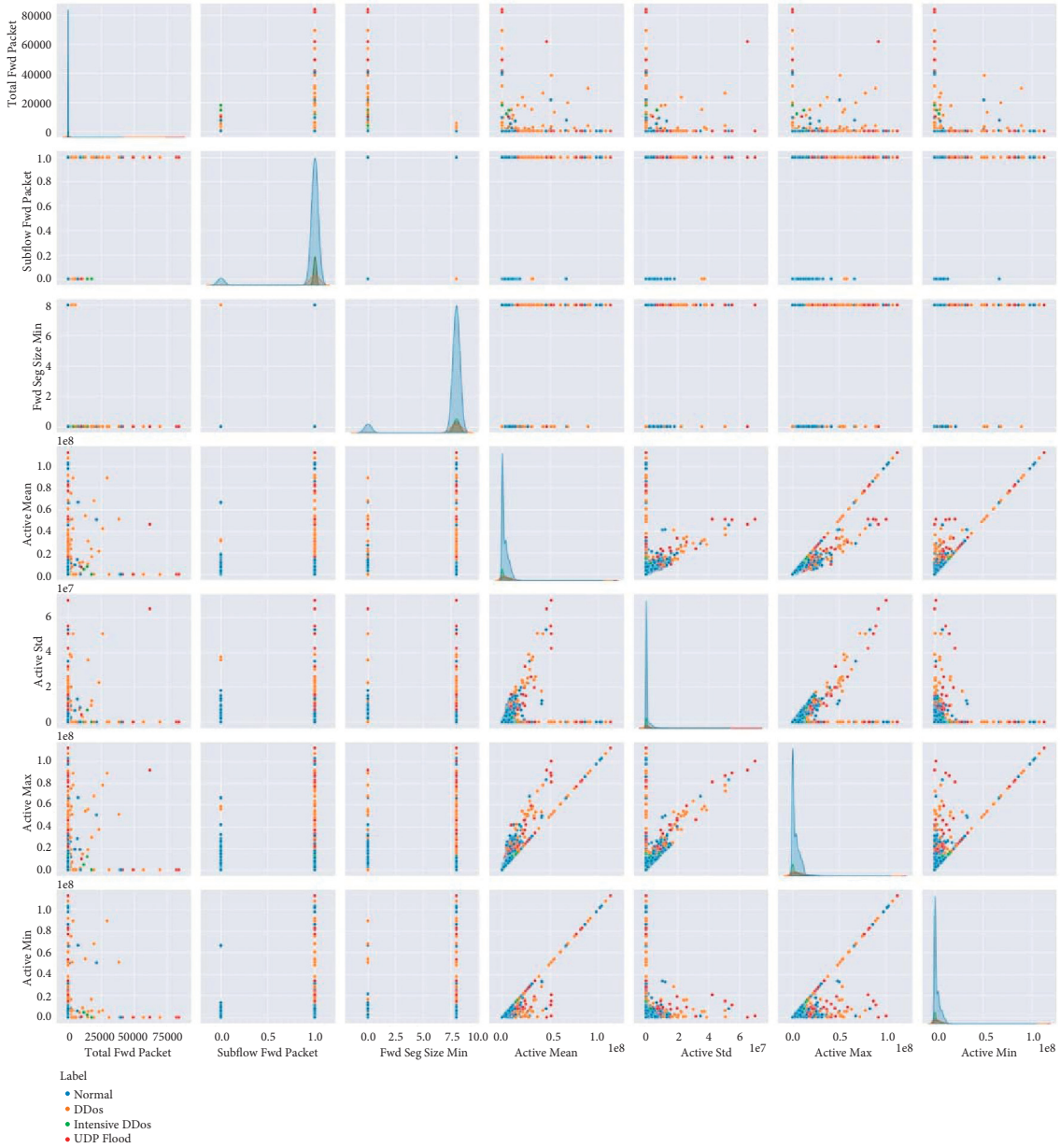


FIGURE 6: Pairwise relationships of the seven best selected features.

The bagging (or Bootstrap Aggregating) method is used to reduce variance and helps to avoid overfitting of machine learning algorithms of multiclass classification. We used all 7 classifiers and concluded that LightGBM classifier shows the best classification results with accuracy of 0.8942.

For the voting method, we selected 5 best classifiers (CatBoostClassifier, XGBClassifier, LGBMClassifier, AdaBoostClassifier, and GradientBoostingClassifier) and considered two voting methods: soft_weight voting with obtained accuracy of 0.8773 and soft_weight voting with obtained accuracy of 0.8933. Soft voting is responsible for the simple averaging of the classifier values in the dataset and the output of average value of the class label. We first evaluated the classifiers separately, and we know their accuracy; from the

obtained accuracy, we know the confidence levels of the classifiers. The confidence level is the weight coefficient; the higher it is for the classifier, the more influence it has on the weighted average value; therefore, the following weights were chosen for the 5 best classifiers: weights = [2, 2, 3, 2, 1].

For the stacking method, we considered the same 5 best classifiers and selected DecisionTreeClassifier and RandomForestClassifier as metaclassifiers. Calculated metrics of ensemble classifiers for both balanced and imbalanced datasets are presented in Figures 10 and 11. RandomForestClassifier showed the best classification results across all researched methods with accuracy of 0.9256; its confusion matrix is shown in Figure 12.

Due to the fact that we use our personal generated dataset, we cannot make a direct comparison with existing

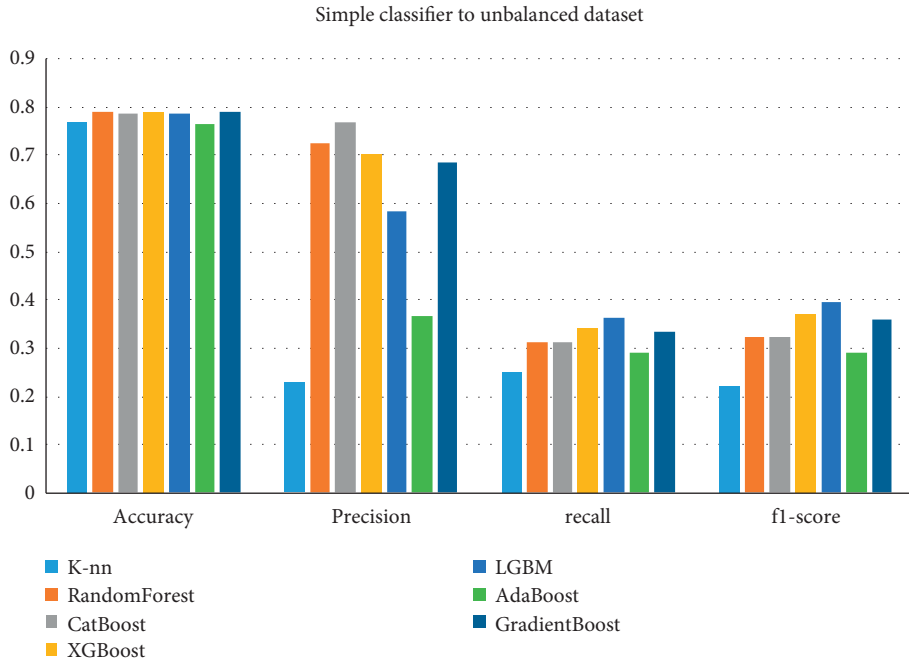


FIGURE 7: Separate classifiers’ metrics comparison on the imbalanced dataset Data.

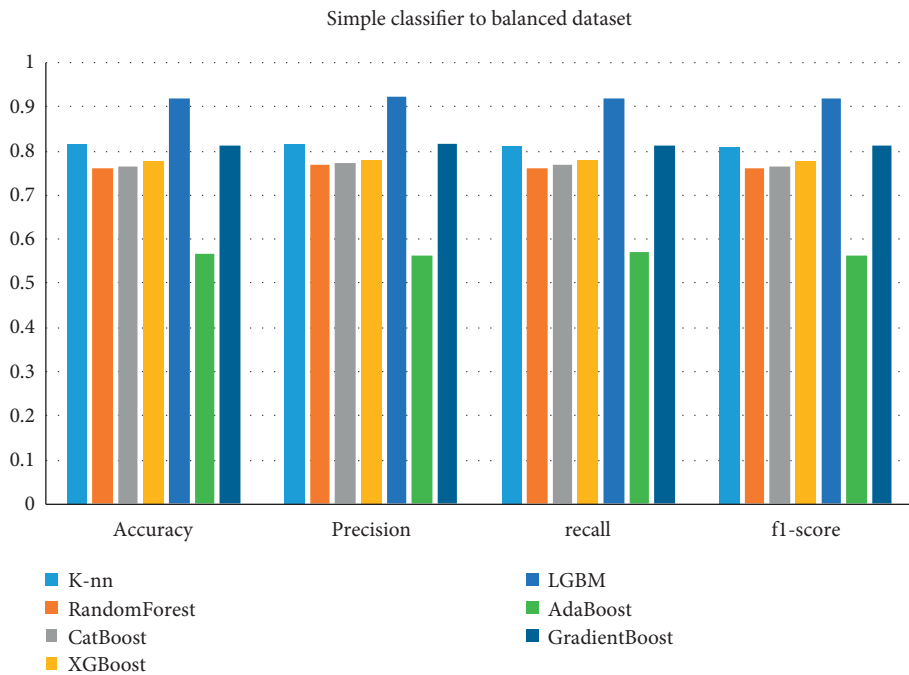


FIGURE 8: Separate classifiers’ metrics comparison on the balanced dataset DataSMOTE.

results in DDoS attack detection. The primary goal of this paper was to demonstrate in detail the process of researching VANET network security issues, from generation of a network segment to usage of up-to-date classifiers and ensembles of classifiers for the multiclass classification problem.

The approach applied in this work to the formation of intelligent models for identifying network attacks can be replaced by an end-to-end pipeline machine learning system. The problem of automated application of machine learning methods to real-world problems is called AutoML. The AutoML [54] pipeline typically includes the

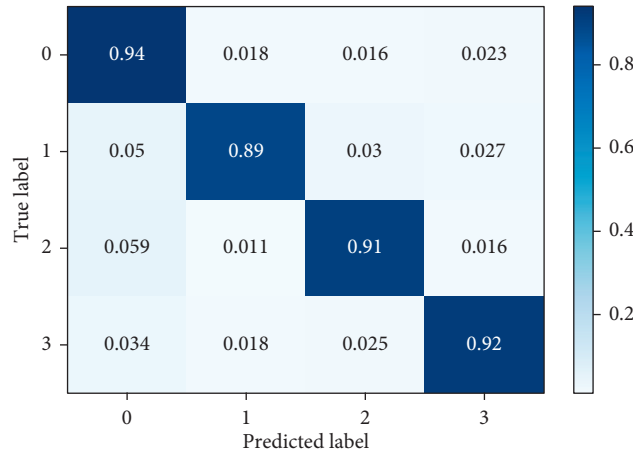


FIGURE 9: Confusion matrix of LightGBM classifier on the balanced dataset DataSMOTE.

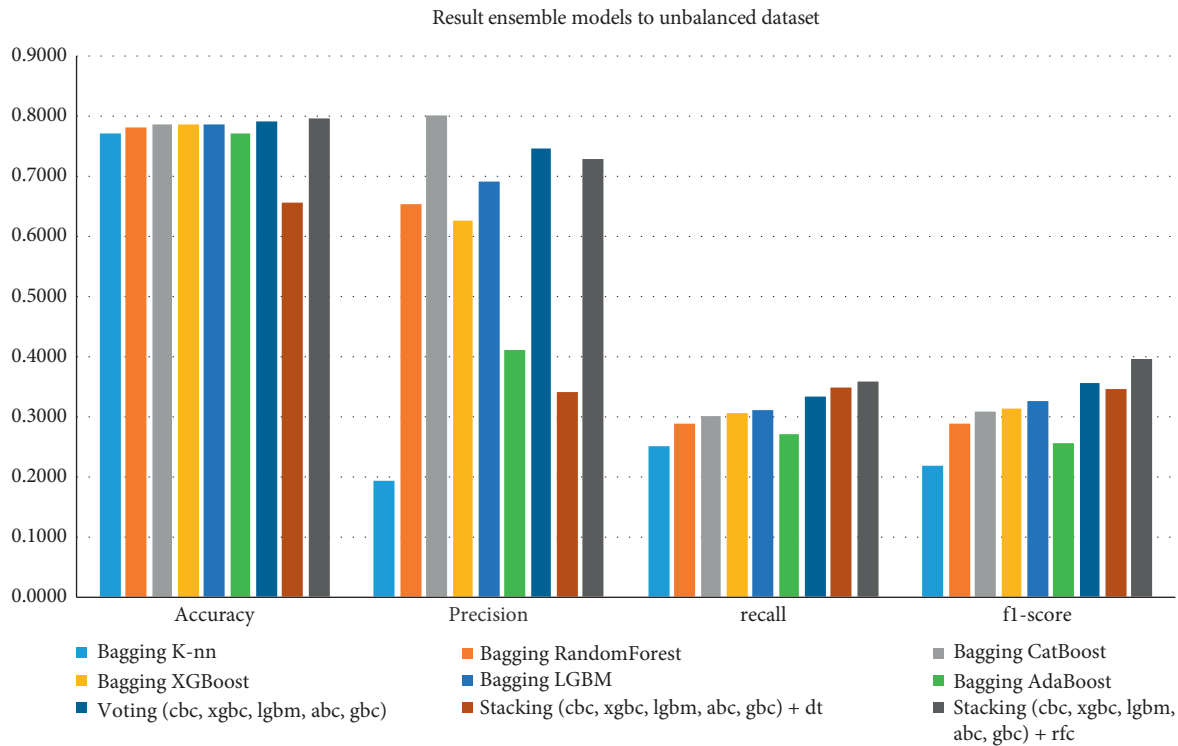


FIGURE 10: Ensemble classifiers’ metrics comparison on the imbalanced dataset Data.

steps of data preparation, feature construction, model generation, and model evaluation. At the moment, there are many libraries that partially solve this problem;

however, working with various kinds of raw data introduces great uncertainty into the assessment of their effectiveness.

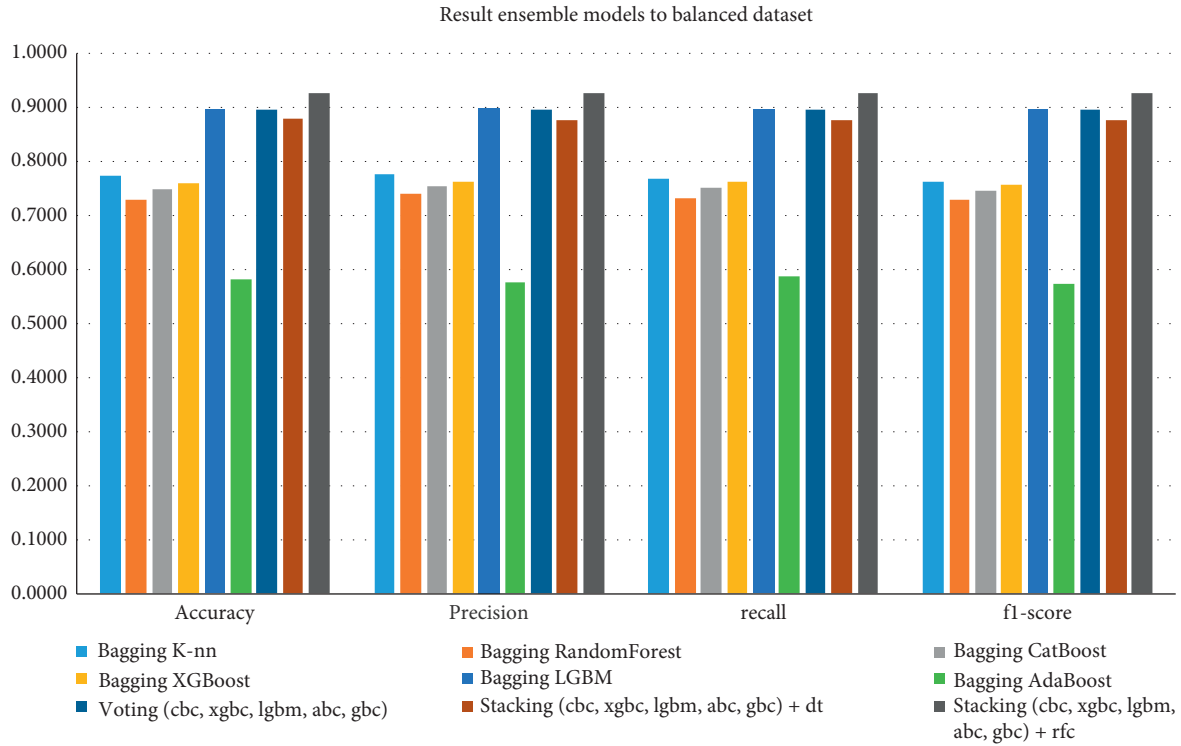


FIGURE 11: Ensemble classifiers’ metrics comparison on the balanced dataset DataSMOTE.

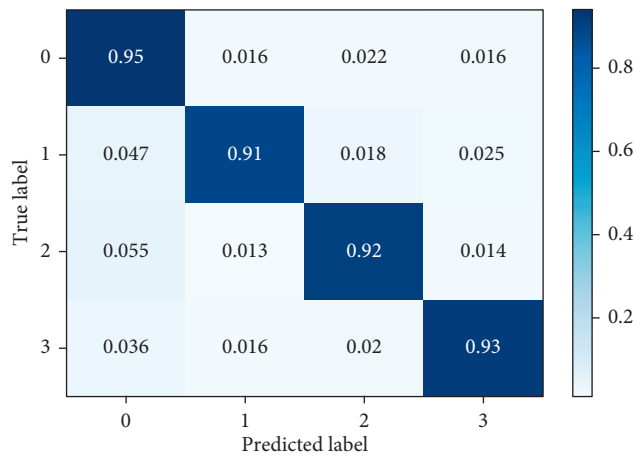


FIGURE 12: Confusion matrix of RandomForestClassifier on the balanced dataset DataSMOTE.

6. Conclusion

OMNeT++ software and INET framework are powerful tools for simulation of mobile ad hoc networks and implementation of various network threats. Modeling of VANET segment was conducted using OMNeT++, and the resulting VANET dataset contains 11212 network traffic flows with 58 extracted features and 3 network attacks behaviors implementations. The obtained dataset is imbalanced; therefore, for further research, balanced dataset was built using SMOTE technique.

An experimental comparison of the quality of modern machine learning methods of multiclass classification on the original and balanced datasets was carried out. The best results with accuracy of 0.9256 were shown by the stacking technique of classifiers with random forest as a metaclassifier.

In future research, we plan to build more simulation scenarios with increased number of vehicles, new mobility movements options, and different network attacks implementation. Our main goal is to increase efficiency of well-known multiclass classification algorithms on arbitrarily generated VANET datasets.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

The work was supported by the National Natural Science Foundation of China (Nos. 61872001 and 62011530046), the Cooperation and Exchange Project between NSFC and RFBR (No. 20-57-53019), and the grant of the President of the Russian Federation (MK-2959.2021.1.6), as well as scholarships of the President of the Russian Federation to young scientists and postgraduates (SP-3652.2021.5), the Open Fund of Key Laboratory of Embedded System and Service Computing (Tongji University), Ministry of Education (No. ESSCKF2018-03), the Open Fund for Discipline Construction (Institute of Physical Science and Information Technology, Anhui University), and the Excellent Talent Project of Anhui University.

References

- [1] C.-X. Wang, M. D. Renzo, S. Stanczak, S. Wang, and E. G. Larsson, "Artificial intelligence enabled wireless networking for 5G and beyond: recent advances and future challenges," *IEEE Wireless Communications*, vol. 27, no. 1, pp. 16–23, 2020.
- [2] J. Cheng, G. Yuan, M. Zhou et al., "Accessibility analysis and modeling for IoV in an urban scene," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 4, pp. 4246–4256, 2020.

- [3] H. Zhong, J. Ni, J. Cui, J. Zhang, and L. Liu, "Personalized location privacy protection based on vehicle movement regularity in vehicular networks," *IEEE Systems Journal*, 2021.
- [4] D. He, S. Zeadally, N. Kumar, and W. Wu, "Efficient and anonymous mobile user authentication protocol using self-certified public key cryptography for multi-server architectures," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 9, pp. 2052–2064, 2016.
- [5] A. Shamir, "Identity-based cryptosystems and signature schemes," *Workshop on the Theory and Application of Cryptographic Techniques*, Springer, Berlin, Heidelberg, Germany, 1984.
- [6] J. Jinyuan Sun, C. Zhang, Y. Zhang, and Y. Fang, "An identity-based security system for user privacy in vehicular ad hoc networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 21, no. 9, pp. 1227–1239, 2010.
- [7] S. F. Tzeng, S. J. Horng, T. Li, X. Wang, P. H. Huang, and M. Khurram, "Enhancing security and privacy for identity-based batch verification scheme in VANETs," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 4, pp. 3235–3248, 2015.
- [8] X. Hu, J. Wang, H. Xu, Y. Liu, and X. Zhang, "Secure and pairing-free Identity-based batch verification scheme in vehicle ad-hoc networks," in *Proceedings of the International Conference on Intelligent Computing*, July 2016.
- [9] L. Wei, J. Cui, Y. Xu, J. Cheng, and H. Zhong, "Secure and lightweight conditional privacy-preserving authentication for securing traffic emergency messages in VANETs," *IEEE Transactions on Information Forensics and Security*, vol. 16, pp. 1681–1695, 2020.
- [10] A. Riyami, S. Sattam, and K. G. Paterson, "Certificateless public key cryptography," in *Proceedings of the International Conference on the Theory and Application of Cryptology and Information Security*, vol. 2894, November 2003.
- [11] A. Malip, S. L. Ng, and Q. Li, "A certificateless anonymous authenticated announcement scheme in vehicular ad hoc networks," *Security and Communication Networks*, vol. 7, no. 3, pp. 588–601, 2014.
- [12] J. Cui, J. Zhang, H. Zhong, R. Shi, and Y. Xu, "An efficient certificateless aggregate signature without pairings for vehicular ad hoc networks," *Information Sciences*, vol. 451–452, pp. 1–15, 2018.
- [13] Y. Ming and X. Shen, "PCPA: a practical certificateless conditional privacy preserving authentication scheme for vehicular ad hoc networks," *Sensors*, vol. 18, no. 5, p. 1573, 2018.
- [14] J. Cui, J. Zhang, H. Zhong, and Y. Xu, "SPACF: a secure privacy-preserving authentication scheme for VANET with cuckoo filter," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 11, pp. 10283–10295, 2017.
- [15] J. Cui, J. Chen, H. Zhong, J. Zhang, and L. Liu, "Reliable and Efficient Content Sharing for 5G-Enabled Vehicular Networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, 2020.
- [16] L. Wei, J. Cui, H. Zhong, Y. Xu, and L. Liu, "Proven secure tree-based authenticated key agreement for securing V2V and V2I communications in VANETs," *IEEE Transactions on Mobile Computing*, vol. 14, 2021.
- [17] N. B. Gayathri, G. Thumbur, P. V. Reddy, and M. Z. Ur Rahman, "Efficient pairing-free certificateless authentication scheme with batch verification for vehicular ad-hoc networks," *IEEE Access*, vol. 6, pp. 31808–31819, 2018.
- [18] M. R. Asaar, M. Salmasizadeh, W. Susilo, and A. Majidi, "A secure and efficient authentication technique for vehicular ad-

- hoc networks,” *IEEE Transactions on Vehicular Technology*, vol. 67, no. 6, pp. 5409–5423, 2018.
- [19] Miracl, “Multiprecision integer and rational arithmetic cryptographic (MIRACL) library,” 2019, <https://github.com/miracl/MIRACL>.
- [20] J. Montenegro, C. Iza, and M. Aguilar Igartua, “Detection of position falsification attacks in VANETs applying trust model and machine learning,” in *Proceedings of the 17th ACM Symposium on Performance Evaluation of Wireless Ad Hoc*, November 2020.
- [21] P. K. Singh, S. Gupta, R. Vashistha, S. K. Nandi, and S. Nandi, “Machine learning based approach to detect position falsification attack in vanets,” in *Proceedings of the International Conference on Security & Privacy*, April 2019.
- [22] F. A. Ghaleb, F. Saeed, M. Al-Sarem et al., “Misbehavior-aware on-demand collaborative intrusion detection system using distributed ensemble learning for VANET,” *Electronics*, vol. 9, no. 9, p. 1411, 2020.
- [23] T. Nandy, R. M. Noor, M. Y. I. B. Idris, and S. Bhattacharyya, “T-BCIDS: trust-based collaborative intrusion detection system for VANET,” in *Proceedings of the National Conference on Emerging Trends on Sustainable Technology and Engineering Applications (NCETSTE)*, February 2020.
- [24] S. Sharma and A. Kaul, “Hybrid fuzzy multi-criteria decision making based multi cluster head dolphin swarm optimized IDS for VANET,” *Vehicular Communications*, vol. 12, pp. 23–38, 2018.
- [25] T. Zhang and Q. Zhu, “Distributed privacy-preserving collaborative intrusion detection systems for VANETs,” *IEEE Transactions on Signal and Information Processing over Networks*, vol. 4, no. 1, pp. 148–161, 2018.
- [26] D. Zhang, F. Richard Yu, and R. Yang, “A machine learning approach for software-defined vehicular ad hoc networks with trust management,” in *Proceedings of the IEEE Global Communications Conference (GLOBECOM)*, December 2018.
- [27] V. Belenko, V. Krundyshev, and M. Kalinin, “Synthetic datasets generation for intrusion detection in VANET,” in *Proceedings of the 11th International Conference on Security of Information and Networks*, Cardiff, UK, September 2018.
- [28] P. K. Singh, R. R. Gupta, S. K. Nandi, and S. Nandi, “Machine learning based approach to detect wormhole attack in VANETs,” in *Proceedings of the Workshops of the International Conference on Advanced Information Networking and Applications*, March 2019.
- [29] S. Kumar, K. Singh, S. Kumar, O. Kaiwartya, Y. Cao, and H. Zhou, “Delimitated anti jammer scheme for Internet of vehicle: machine learning based security approach,” *IEEE Access*, vol. 7, pp. 113311–113323, 2019.
- [30] A. Rehman, S. Rehman, M. U. Khan, M. Alazab, and T. Reddy, “CANintelliIDS: detecting in-vehicle intrusion attacks on a controller area network using cnn and attention-based gru,” *IEEE Transactions on Network Science and Engineering*, vol. 8, no. 2, 2021.
- [31] R. Jhaveri, S. V. Ramani, G. Srivastava, T. R. Gadekallu, and V. Agarwall, “Fault-resilience for bandwidth management in industrial software-defined networks,” *IEEE Transactions on Network Science and Engineering*, vol. 8, 2021.
- [32] N. Akhtar, S. C. Ergen, and O. Ozkasap, “Vehicle mobility and communication channel models for realistic and efficient highway VANET simulation,” *IEEE Transactions on Vehicular Technology*, vol. 64, no. 1, pp. 248–262, 2014.
- [33] Sumo, “simulation of urban MObility,” 2001, <http://sumo.sourceforge.net>.
- [34] F. Michaeler and C. Olaverri-Monreal, “3D driving simulator with VANET capabilities to assess cooperative systems: 3DSimVanet,” in *Proceedings of the 2017 IEEE Intelligent Vehicles Symposium (IV)*, June 2017.
- [35] D. S. Buse, “Christoph Sommer, and Falko Dressler. Demo abstract: integrating a driving simulator with city-scale VANET simulation for the development of next generation ADAS systems,” in *Proceedings of the 2018-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPs)*, April 2018.
- [36] C. Obermaier, R. Riebl, and C. Facchi, “Fully reactive hardware-in-the-loop simulation for vanet devices,” in *Proceedings of the 2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, November 2018.
- [37] OMNeT, “OMNeT++Discrete event simulator,” 2020, <https://omnetpp.org/>.
- [38] T. O. Fahad and A. A. Ali, “Compressed fuzzy logic based multi-criteria AODV routing in VANET environment,” *International Journal of Electrical and Computer Engineering*, vol. 9, no. 1, p. 397, 2019.
- [39] B. P. Maratha, T. R. Sheltami, and K. Salah, “Performance study of MANET routing protocols in VANET,” *Arabian Journal for Science and Engineering*, vol. 42, no. 8, pp. 3115–3126, 2017.
- [40] Network world, “NCTUns 6.0 Network Simulator and Emulator,” 2017, <http://nsl.cs.nctu.edu.tw/NSL/nctuns.html/>.
- [41] C. Raj, T. Makwana’s, U. Upadhayaya, and P. Mahida, “Simulation of VANET using ns-3 and SUMO,” *International Journal of Advanced Research in Computer Science and Software Engineering*, vol. 4, p. 4, 2014.
- [42] NS-3, “Discrete-event network simulator,” 2010, <https://www.nsnam.org/>.
- [43] S. So, P. Sharma, and J. Petit, “Integrating plausibility checks and machine learning for misbehavior detection in VANET,” in *Proceedings of the 2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*, December 2018.
- [44] J. Grover, N. K. Prajapati, V. Laxmi, and M. S. Gaur, “Machine learning approach for multiple misbehavior detection in VANET,” in *Proceedings of the International Conference on Advances in Computing and Communications*, July 2011.
- [45] Yi Zeng, Z. Ming, and M. Liu, “Senior2local: a machine learning based intrusion detection method for vanets,” in *Proceedings of the International Conference on Smart Computing and Communication*, December 2018.
- [46] K. Sharshembiev, S. M. Yoo, and E. Elmahdi, “Protocol misbehavior detection framework using machine learning classification in vehicular Ad Hoc networks,” *Wireless Networks*, vol. 27, pp. 1–16, 2021.
- [47] B. A. Tama and S. Lim, “Ensemble learning for intrusion detection systems: a systematic mapping study and cross-benchmark evaluation,” *Computer Science Review*, vol. 39, Article ID 100357, 2021.
- [48] L. Zhang, Q. Wu, J. D. Ferrer, B. Qin, and C. Hu, “Distributed aggregate privacy-preserving authentication in VANETs,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 3, pp. 516–526, 2016.
- [49] INET, “INET Framework,” 2002, <https://inet.omnetpp.org/>.
- [50] C. Sommer, R. German, and F. Dressler, “Bidirectionally coupled network and road traffic simulation for improved IVC analysis,” *IEEE Transactions on Mobile Computing*, vol. 10, no. 1, pp. 3–15, 2010.

- [51] L. A. Habibi, G. D. Gil, M. Mamun, and A. A. Ghorbani, "Characterization of tor traffic using time based features," *ICISSp*, 2017.
- [52] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: synthetic minority over-sampling technique," *Journal of Artificial Intelligence Research*, vol. 16, pp. 321–357, 2002.
- [53] C. Bentéjac, C. Anna, and G. Martínez-Muñoz, "A comparative analysis of gradient boosting algorithms," *Artificial Intelligence Review*, vol. 54, pp. 1–31, 2020.
- [54] Autogluon, "AutoML for text, image, and tabular data," <https://github.com/awsmlabs/autogluon>.

Research Article

Research on Medical Waste Supervision Model and Implementation Method Based on Blockchain

Hui Wang,¹ Longshuai Zheng,² Qihong Xue,² and Xueqing Li ¹

¹School of Software, Shandong University, Jinan 250101, Shandong, China

²Institute of Software, Chinese Academy of Sciences, Beijing 100190, China

Correspondence should be addressed to Xueqing Li; xqli@sdu.edu.cn

Received 9 September 2021; Revised 6 January 2022; Accepted 10 January 2022; Published 24 February 2022

Academic Editor: Lu Liu

Copyright © 2022 Hui Wang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The Internet of Things (IoT) has brought unprecedented changes to the society and permeated our daily life. However, it has not been successfully applied in the area of medical waste regulation, where the recycling and disposal of medical waste have significant loopholes in the management of classification, transportation, disposal, supervision, and other links. The source, authenticity, and integrity of medical waste data lack guarantees, and there is a risk of data tampering and forgery. Although there are currently some medical waste supervision applications combined with IoT-based blockchain domestically and internationally to facilitate information sharing and transfer, no verifiable method is provided for the information privacy leakage of medical waste operators. To address this problem, we propose a blockchain-based medical waste supervision model, which connects participants involved in the process, introduces digital credentials to achieve the protection of operator information privacy, and ensures that the entire data process is authentic and credible. By building a decentralized system architecture and setting intelligent contracts, we integrate and record the medical waste disposal regulatory information in different phases on the blockchain to form the supervision of medical waste chain. In addition, we digitize the physical credentials and certificates using digital credentials to achieve cryptography security and privacy protection. The regulatory model designed in this paper can provide digital certificates of disposal tracking information to the health, environmental protection, and other administrative departments in China. It can provide authoritative evidence for the supervision and accountability of medical waste disposal and support the construction of a new generation of medical waste regulatory information systems in China.

1. Introduction

Medical waste, known as “No. 1 hazardous waste,” refers to the waste containing directly or indirectly infectious, viral, and other hazards produced by medical and health institutions in medical treatment, prevention, healthcare, and other related activities [1]. China has banned the sale of medical waste for many years. However, driven by interests, individuals still make infusion tubes into plastic raw materials and even make illegal profits by reselling placentas. There are large loopholes in the management of classification, transportation, disposal, and supervision of medical waste recycling and disposal [2]. For example, when the waste handover is recorded manually, the query and statistics work are complicated. It is difficult for managers to

monitor this work in real time, and it is difficult to trace the responsibility subject.

To manage medical waste more safely, in July 2018, the National Health Commission launched the pilot application construction project of the provincial credit information management platform and selected Fujian Province and other pilot provinces to build a medical waste supervision system based on IoT-enabled blockchain technology. In 2020, Shandong, Shaanxi, and other provinces were added to continue to expand the scale of this application. With the outbreak of COVID-19, the National Health Commission has increasingly stringent regulatory requirements for medical waste, and new solutions are urgently needed to improve the efficiency of supervision.

The informatization construction, also known as IoT technology application, helps to improve the standardization in the whole medical waste recycling and disposal [3] process. The application of IoT has made an apparent shift and contributed to operation optimization for all industries, including agriculture [4] and transportation [5]. RFID-enabled or Internet-based devices are connected and organized as an information network, making it efficient and productive to integrate trackable data. IoT has the advantages of real-time performance and all-in-one efficiency that enable users to manage and supervise physically isolated devices as a system.

However, such information-based transformation schemes generally transfer the offline data to online through intelligent devices such as code scanners [4]. This transformation scheme is convenient for the query and statistical analysis of relevant data, but there is no innovation in the management mode and management manners of medical waste. There are still some issues in medical waste recovery and disposal, such as shortage of weight, black-box operation, and tampering with credentials.

Furthermore, integrating IoT with blockchain technology enhances the transparency and credibility of the management process. Blockchain is a distributed ledger technology that combines distributed data storage, point-to-point transmission, consensus, and encryption algorithms [5–9]. Encryption algorithm, blockchain structure, and alliance consensus [10] ensure the authenticity, integrity, and nontampering with data on the chain. Integrating blockchain into the management of the whole process of medical waste recovery and disposal can reduce the risk of data tampering and falsification [11, 12], guarantee the safety of medical waste data, and improve the supervision level of medical waste.

Current research on medical waste management using IoT and blockchain focuses on the following aspects. (1) Store the business credentials of each link of recovery and disposal on the blockchain to ensure that the data cannot be tampered with [13]. (2) Unified medical waste management, transport vehicles, and people are realized through joint IoT technology [14, 15]. Verify the validity of clinical waste recovery and disposal process through smart contract [16]. (3) Introduce the token reward and punishment mechanism to force participating entities to comply with medical waste disposal rules and other aspects [17]. However, there is a lack of lifecycle management of medical waste, construction of the alliance of stakeholders, and privacy protection mechanism of the participants in the core link.

In terms of stakeholder decentralized collaboration and privacy protection, the proposed general solution, whether in academia or industry, has been one of the issues focused on for a long time. Once a broad and effective solution can be designed under a reasonable security assumption, its impact is obvious. The technologies that are expected to be close to general solutions currently mainly include secure multiparty calculations based on computational difficulty theory, homomorphic ciphertext calculations, and zero-knowledge proofs. However, the above three types of technologies have significant practical limitations. The industry has also tried

to rely on a trusted hardware execution environment to build general solutions, but the effectiveness of its actual privacy protection is difficult to verify publicly.

Blockchain technology is not only a decentralized collaboration solution, but as an effective privacy protection solution, it is promoted by introducing breakthrough optimization factors such as sociology, psychology, and economics principles; rational participant models; and multiparty incentive mechanisms. The balance between insensitive user experience and effective privacy protection approaches Pareto optimality. In the case of medical waste management, blockchain systems can provide medical waste management platforms for the collection of waste in cities to maximize sustainability level in terms of health, social, environmental, and financial aspects. The data and transactions stored on the blockchain are accessible to the stakeholders involved in the forward supply chain and waste management processes through distributed public or private ledgers. The decentralization feature of blockchain increases the trust among stakeholders as it eliminates the need to assess the trustworthiness of the participants.

In this paper, we design a medical waste supervision model based on IoT and blockchain techniques to address the above problems. This model integrates blockchain technology into the generation, transportation, treatment, and destruction of medical waste. By connecting different stages of waste disposal, defining the critical node data on the blockchain, and mutually verifying the data under the blockchain, the security and immobility of the data are ensured while achieving system efficiency. In addition, digital credentials are used to enable users to disclose information to protect the privacy of handover personnel selectively. Moreover, on-net monitoring of the flow of medical waste data allows assessing and detecting medical waste trading violations.

1.1. The Main Contributions of This Article. Compared with the traditional medical waste supervision methods, our proposed IoT and blockchain-based medical waste supervision model has the following advantages:

- (1) Data security: the critical data of the medical waste treatment process is linked to a certificate to ensure authenticity and nontampering.
- (2) Multiparty participation: the decentralized architecture allows the application scenario of multirole subjects by compartmentalizing the corresponding mission, while taking into account both efficiency and security.
- (3) Clear rights and responsibilities: relevant responsible persons and operators need to sign when submitting data to the blockchain, and the time, place, person, operation process, and result of data collection are linked at the same time.
- (4) Privacy protection: the IoT operator's private information is hidden through the digital credential, and a unique identifier is given on the blockchain to represent each operator; the factual personnel

information is stored in the digital certificate, and all parties can verify its authenticity in the process of circulation to prevent privacy disclosure.

In terms of the application and practical value of this model, we developed a medical waste supervision system based on the regulatory model proposed in this paper, the RepChain blockchain essential components [18, 19] independently developed by the Institute of Software of the Chinese Academy of Sciences, and related technologies of the IoT [20]. The system has been applied in hospitals, transportation, environmental protection, and disposal institutions in Shandong, Fujian, and other places in China. It can improve management methods, effectively reduce regulatory costs, improve regulatory efficiency, and achieve good results.

1.2. The Organizational Structure of This Article. Section 2 introduces related works and discusses the traceability or supervision of medical waste and other wastes utilizing IoT-based blockchain technology. Section 3 proposes the medical waste supervision model based on reproducing the traditional medical waste management process, including the digital evidence and the blockchain deployment model. Section 4 introduces the medical waste supervision system. Section 5 combines the current situation of the medical waste industry in Shaanxi Province, develops the relevant condition of the medical waste supervision system according to this model, and evaluates the outcome. Section 6 summarizes the paper.

2. Related Works

2.1. Background. Technology-driven methods for medical waste management generally use bar codes, two-dimensional codes, RFID, and other [21, 22] IoT-related methods to track and realize the traceability of the whole process of medical waste management. However, they are all based on forming a strictly closed loop in the collection and transportation process. In reality, medical institutions, transport agencies, and disposal agencies have their waste information management platform. The information management systems among medical institutions are also uneven, so it is difficult for the regulatory authorities to carry out unified and effective management. Blockchain-based medical waste management aims to connect these separated systems and create a traceable and transparent, automated rule engine to solve openness, interoperability, and decentralization of medical waste.

2.2. The Research Status at Home and Abroad. In the research of “blockchain + medical waste disposal,” Li et al. proposed the integrated development of blockchain and medical waste management. Medical waste disposal operators of each process store relevant information on the blockchain on time. They rely on the tracing source code as the carrier of information transmission, through the tracing source code collection and monitoring data, and tracking and confirming

the treatment, transportation, disposal, and other links of medical waste to achieve multidimensional network supervision. In terms of alliance building, it is proposed that hospitals, cleaning companies, the public, and national supervisory agencies participate in the decentralized autonomous organization simultaneously and use the interests of all parties to play a checking role [23]. Ahmad et al. [24] proposed integrating Ethereum and interplanetary file system (IPFS) for the supply chain for COVID-19 medical devices to securely access, store, and share data related to COVID-19 medical devices and their waste management. They also define rules of interaction for waste disposal so that governments can impose penalties on stakeholders if violations occur. Kassou et al. [1] show that medical wastewater is based on blockchain technology and the IoT system to effectively manage, coordinate, and monitor medical wastewater, such as in-hospital using flow meter, water meter, and intelligent Internet of sensing equipment. It will produce the data using blockchain for the witness, to government agencies and stakeholders involved in the common node, which improves the process brightness and supervision.

In patent research and development, Jiang and Tian [25] proposed a blockchain-based medical waste supervision platform and management method. They built a blockchain-based medical waste whole-lifecycle management method in medical institutions through the comprehensive application of blockchain, big data, and IoT technologies to eliminate regulatory gaps and blind areas. Their way solves the long-standing problems of medical institutions, health commissions, and environmental protection departments that cannot be traced, collected, and held accountable. Lin [26] proposed to connect the remote central regulatory server with the device for local treatment of medical waste, the intelligent medical waste collection vehicle, the legal person classification collection bag of the medical department, and the collection and transfer device for disposal of medical waste. Then, the IoT and blockchain were used to achieve intelligent supervision of the whole process, to achieve hierarchical regulation and hierarchical statistical check.

There are few studies on the application of blockchain in the field of medical waste. In terms of sorting out appropriate research methods, it is extended to the area of “blockchain + e-waste or solid waste” management. Gupta and Bedi [27] proposed an e-waste management system based on Ethereum, which considers the main stakeholders, including electronic component manufacturers, consumers, and retailers, to ensure that participants comply with the e-waste disposal guidelines. According to the defined abnormal events, it uses blockchain to record, report, and verify all electronic waste sold and uses smart contracts to impose penalties on those who are responsible. Laouar et al. [28] proposed the continuous monitoring and tracking of municipal solid waste transportation participants, responsible persons, transportation tracks, collectors, processors, etc. They stored the QR (quick response) codes generated by the data for solid waste identification on the blockchain. Solid waste vehicles’ state, location, and routing information are managed using an off-chain storage system to balance safety and throughput.

2.3. Research Focus of This Article. To sum up, the current IoT-based and blockchain-based medical waste management related studies focus on the following aspects. (1) Save the business credentials of each link of recovery and disposal on the blockchain to ensure data security. (2) Unified medical waste management, transport vehicles, and people are realized through joint IoT. (3) Verify the validity of the clinical waste recovery and disposal process through smart contracts. The current research lacks a mechanism for the whole-lifecycle management of medical waste and the privacy protection of participants in the core link. In contrast, this paper proposes a comprehensive medical waste regulatory model, which combines the domestic medical waste management process with IoT and blockchain technologies to build a multirole entity alliance. In the case of the whole-process tracking of the responsible subject, the privacy and security of operators are protected. The result is effective regulation of medical waste in a safe, transparent, trusted, decentralized, and auditable manner.

3. Medical Waste Supervision Model Based on Blockchain

In this section, the proposed model for IoT-based and blockchain-based medical waste supervision is introduced. The goal of our system is to ensure transparent and explicit assignment of responsibilities and credible information management for flexible regulation. We demonstrate the process of waste disposal and how we integrate mentioned techniques to guarantee credibility and efficiency. Digital credential is used to assure security and verify authenticity.

3.1. Medical Waste Disposal Process. Based on the Product Lifecycle Management (PLM) [29] model of medical waste and the basic logic of IoT, the whole process of medical waste disposal is supervised from the collection, storage, and transshipment to the disposal and combined with the credit mechanism of blockchain technology and the characteristics of decentralization. A tamper-proof tracking account book for the whole process of medical waste disposal is established to provide users with credible and traceable management information.

This model provides a convenient way for waste disposal personnel to report the data of disposal nodes through smart devices, while providing an aggregated information platform for supervisors to realize the automatic supervision of information by displaying each node of medical waste disposal and monitoring the nodes to promote the timely delivery of medical waste. This process of information collection and supervision improves the standardized management of medical waste disposal, accounts for the medical waste disposal information, and provides a reliable and traceable digital certificate for the supervision and punishment of medical waste disposal. In the disposal process of medical waste, all transaction information occurring on the node is recorded through the contract, so that the process is

highly transparent. The proposed IoT blockchain model is shown in Figure 1.

We distribute the responsibilities of different stages in the waste monitor recycle network where timely and credible data exchange plays an essential role. The material and information flow of medical waste treatment is as follows.

3.1.1. Medical Institutions. Medical personnel classify, collect, and sort medical waste according to the Medical Waste Classification Catalog. Medical waste management professionals transport the classified and packaged clinical waste from the medical waste generation site to the temporary storage room in the hospital according to the specified route on a daily basis. The loss and leakage of medical waste should be prevented during transportation. Medical waste management professionals weigh medical waste at the site where it is generated on a daily basis. Registration includes the source, type, weight, delivery time, final destination, and operator. After the medical waste is transferred out, full-time personnel shall clean and disinfect the temporary storage sites and social facilities in a timely manner and make good records.

3.1.2. Transport Company. When collecting clinical waste, the transport personnel of the transport company should register the waste and sign their names. The content of the registration includes the place where clinical waste is generated, the date, the type of waste, and the matters to be explained. The registration transfer form is kept on file at the Medical Waste Management Office for three years. When transporting medical waste, it is necessary to prevent the damage of medical waste containers; the loss, leakage, and diffusion of medical waste; and the direct contact of medical waste with the body. All medical waste should be transported in bags and sealed. Crossing medical areas, human activity areas, food processing areas, etc. should be avoided.

3.1.3. Disposal Company. The medical waste must be disposed of by the centralized medical waste disposal unit approved by the ecological environment department. The whole medical waste treatment process can be traced throughout. Different types of medical waste are absorbed in different ways. For example, the treatment of infectious waste is high temperature incineration. The hazardous waste will be incinerated at high temperatures to destroy its shape and tested for toxicity before being buried.

In the medical waste disposal process (see Figure 2), all transaction information that occurs on the node is recorded through the contract, making the process highly transparent:

- (1) The nursing staff and the department staff weigh and encapsulate the medical waste face to face; the nursing staff prints the label, and the department staff scans and confirms the label and transfers the data to the chain.
- (2) The nursing staff transports the medical waste to the temporary storage point and conducts face-to-face

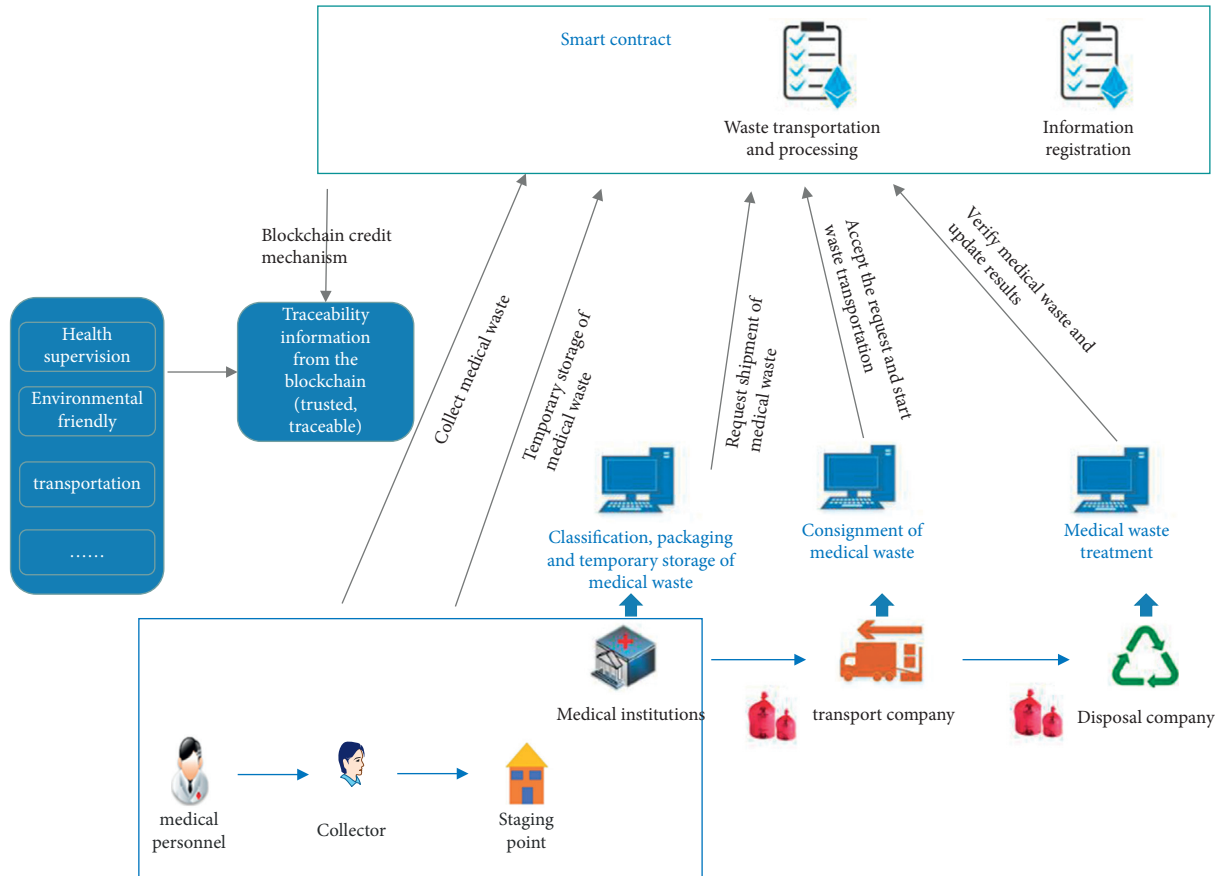


FIGURE 1: The business model of medical waste treatment.

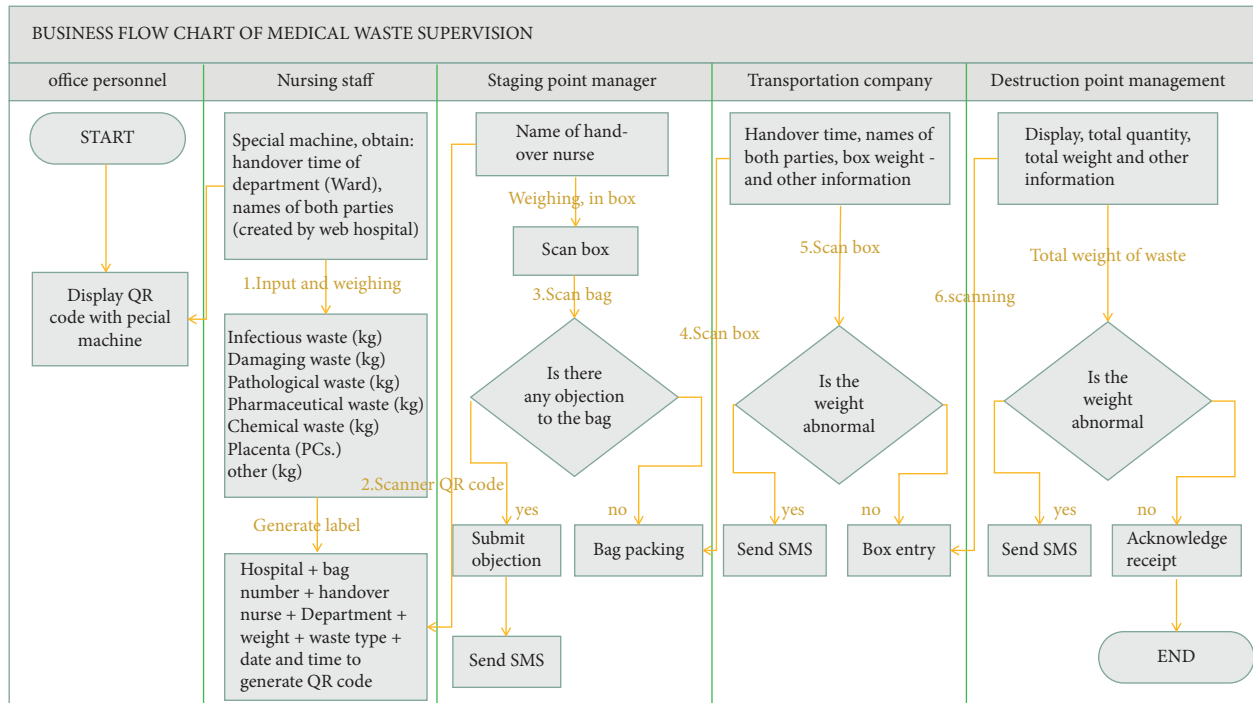


FIGURE 2: Business flow chart of medical waste supervision.

weighing and handing over with the temporary storage point management personnel. After confirming that the weight of the waste is correct, a QR code is generated for the box, and the handover data is uploaded to the chain

- (3) The management personnel of the hospital's temporary storage point carry out the operation of waste classification and scanning into the bin.
- (4) The transportation personnel and the hospital temporary storage point management personnel carry out the weighing and transfer of the boxes. After scanning and confirming that the box weight is correct, the transfer data is uploaded to the chain.
- (5) The transportation personnel and the personnel at the destruction point weigh the vehicle, and after confirming the receipt, the handover data will be uploaded to the chain.

3.2. Digital Credential Model

Definition 1. A digital credential is a document that is digitized and is cryptographically verifiable, presented in a digital form for easy storage and transmission and based on cryptographic mechanisms to more securely and reliably verify its authenticity.

The digital credential model mainly includes the following:

- (i) The certificate issuer, i.e., the medical waste regulatory agency, constructs the digital certificate according to the attribute structure of the digital certificate and is responsible for issuing and updating the certificate.
- (ii) The holder of the credential, i.e., the medical waste operator, applies for the digitization of the credential and verifies its correctness.
- (iii) Credential verifiers, i.e., participants of medical waste, verify the authenticity and validity of the received digital credentials. Participants include health regulatory authorities, medical institutions, transportation departments, transportation companies, environmental protection departments, and disposal companies.
- (iv) Digital credential status includes the following types: valid, revoked, and deactivated.

The digital credential model is illustrated in Figure 3, and an example is shown in Figure 4.

There are many links in the supervision of medical waste, and the privacy disclosure of operators in each link is an essential reason for the relevant personnel to operate in a dark box or engage in dark transactions. To this end, this paper proposes a digital credential model, which hides the operator's private information in the chain, represents each operator by a unique identifier, and stores the basic information in the digital credential.

Operators in all stages of medical waste disposal apply for digital certificates to the regulatory agency. The

regulatory agency issues digital credentials with a unique ID (identification number, identity) and signs the digital credential information based on the asymmetric key's digital signature algorithm. After digitizing the credential, the unique ID of the credential, credential status, and other information are uploaded to the chain and available for information verification by other institutions. The specific information of the operator is not disclosed on the chain (or expressed in the form of a pseudonym) to achieve the purpose of protecting the privacy of personnel information.

3.2.1. Detailed Operation Process

- (1) Applying for digital credentials: medical waste operators need to apply for digital credentials and provide their personal information to the regulatory agency before processing medical waste. The digital credentials are used as a personal identification.
- (2) Issuing digital credentials: the regulatory agency obtains the format and structural attributes of the digital credentials that need to be issued from the blockchain, fills in the corresponding information of the operator according to the format and structural attributes, and signs the credential information.
- (3) Presenting digital credentials: when the operator accesses the services provided by each participant, each participant informs the operator of the credential information that needs to be provided; the operator finds the credential information that meets the needs of each participant from the digital credentials held by the individual, and provides the credential information to each participant for verification.
- (4) Verification of digital credentials: after receiving the credential information of the operator, each participant first verifies the credential information and verifies whether the signature of the presented information is correct; then verifies the correctness of the credential information; and finally obtains it from the blockchain. The validity status of its credentials and corresponding attributes is to determine whether the credentials and corresponding attributes are valid and then determine whether the credential signature information is correct.
- (5) Maintaining status: when a regulatory agency issues a credential, it needs to add its credential status information to the blockchain and initialize it; when it needs to change the status of the credential, such as freezing, restoring, or revoking the credential, it needs to change the corresponding credential.
- (6) Updating attributes: regulators can update certain attribute values of the issued credentials as needed, without revoking the entire credentials, just reissuing the credentials after the update.

An example of the attribute structure information of a digital credential is shown in Figure 5.

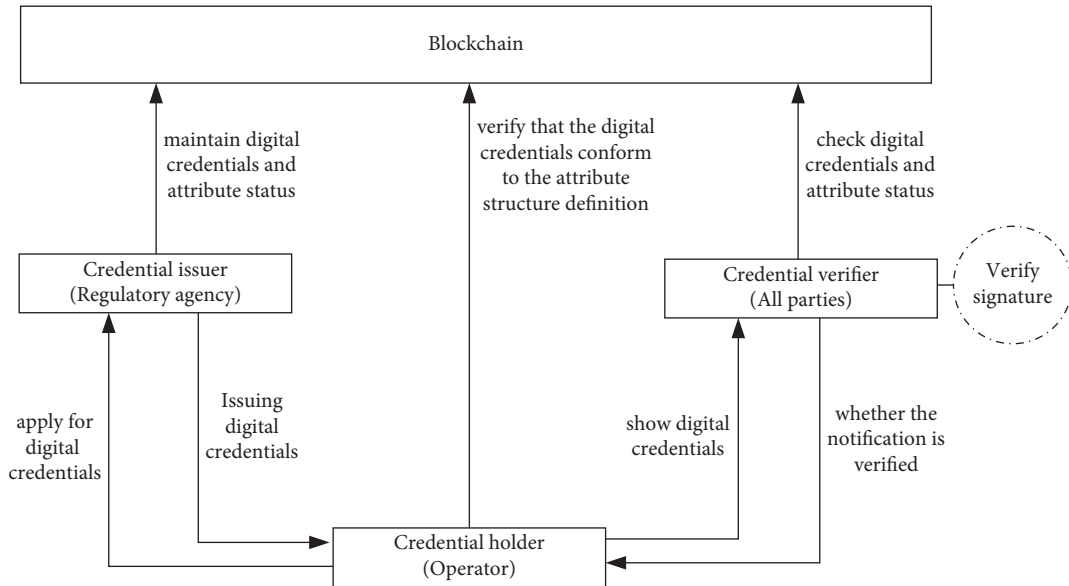


FIGURE 3: Digital credential model.

```

{
  "@context": ["https://www.w3.org/2018/credentials/v1"],
  "id": "medical-0001",
  "type": ["VerifiablePresentation", "medicalwaste"],
  "holder": "did:rep:network_1:9876543",
  "verifiableCredential": {
    "@context": ["https://www.w3.org/2018/credentials/v1"],
    "id": "0123456789abcdef",
    "type": ["VerifiableCredential", "medicalwaste"],
    "claimScheme": "CCS-0001",
    "issuer": "did:rep:network_1:1234567",
    "validFrom": "2021-01-19T19:05:22Z",
    "validUntil": "9999-01-19T19:05:22Z",
    "credentialSubject": {
      "0": { "id": "did:rep:network_1:9876543" },
      "2": { "Name": "Name_001" },
    },
  },
  "proof": {
    "type": "EcdsaSecp256k1Signature2019",
    "created": "2021-01-01T09:05:22Z",
    "verificationMethod": "did:rep:network_1:1234567#key1",
    "signature": {
      "0":
        "gcyudgda898cdbjsdhGYUGHJGBJHDSHJ&867tHJSGHJ.....JUHBXCYUDyt876732btd67120",
      "2": "JGYUIDHJhdscui91289uxcjdsnjkhujkJNXSKJHK.....987238976sgx67gsGHJSGSH"
    }
  }
}
{
  "@context": ["https://www.w3.org/2018/credentials/v1"],
  "id": "0123456789abcdef",
  "type": ["VerifiableCredential", "medicalwaste"],
  "claimScheme": "CCS-0001",
  "issuer": "did:rep:network_1:1234567",
  "validFrom": "2021-01-19T19:05:22Z",
  "validUntil": "9999-01-19T19:05:22Z",
  "credentialSubject": {
    "0": { "id": "did:rep:network_1:9876543" },
    "2": { "Name": "Name_001" },
  },
  "proof": {
    "type": "EcdsaSecp256k1Signature2019",
    "created": "2021-05-01T19:25:23Z",
    "verificationMethod": "did:rep:network_1:9876543#key2",
    "challenge": "xhjgh-8392-ncjds",
    "signature":
      "kdhsHJJKKAuiTRDyqty767w21gygxd16VG91.....hdshhYUGS89789GJGH679GYJGHJGstyaftsa"
  }
}
    
```

FIGURE 4: Example of digital credential presentation.

```

{
  "Unique identification of credential attribute structure": "medical-0001",
  "Credential type name": "mobilenumberCredential",
  "Credential attribute structure version": "0.1",
  "Description of credential attribute structure": "Operator, including detailed information such as unique ID, name, ID number, mobile phone number",
  "Credential attribute structure creation time": "2021-01-18T19:05:22Z",
  "Credential attribute collection": [
    {
      "Attribute name": "id",
      "Type of attribute value": "String",
      "Property description": "ID of the operator"
    },
    {
      "Attribute name": "name",
      "Type of attribute value": "String",
      "Property description": "Name of the operator"
    },
    {
      "Attribute name": "identity-number",
      "Type of attribute value": "String",
      "Property description": "ID number of the operator"
    },
    {
      "Attribute name": "mobile-number",
      "Type of attribute value": "String",
      "Property description": "Mobile phone number of the operator"
    }
  ]
}

```

FIGURE 5: Sample diagram of attribute structure information of a digital credential.

4. Implementation Method of Medical Waste Supervision System Based on Blockchain

This section presents a detailed analysis of the supervision system built based on the concept of real time and security. We use a clear layered structure and apply RepChain as the evidence chain connecting related practitioners and organizations. The specific implementation method is described as follows.

4.1. Architecture of Medical Waste Supervision Blockchain System. The medical waste supervision framework comprises the basic layer of the alliance chain platform and the application layer of medical waste supervision. The former provides IoT-based blockchain services, and the latter provides supervision services for medical waste. From the bottom to the upper layer, there are the data storage, component, application interface, and medical waste supervision application layers, which guarantees the data transfer, as shown in Figure 6.

- (1) Data storage layer: it is divided into file storage and database storage, wherein the former mainly stores block segmented files and the latter is mainly used to store blocks, transaction indexes, and contract status.
- (2) Component layer: it provides network transmission, verification mechanism, contract running engine, consensus mechanism, and other components and provides basic services for the application interface layer.

- (3) Application interface layer: it provides an external interface to interact with the blockchain system in the form of RESTful API; the application interface layer provides basic functions such as transaction submission, and transaction and block retrieval.
- (4) Application layer: in combination with the business process of medical waste supervision, the management of contracts, interfaces, and digital credentials is realized based on the application interface layer from the four aspects of collection, storage, transportation, and disposal.

4.2. Stakeholder Alliance and Consensus Node Construction. In the deployment of blockchain, the autonomous and controllable blockchain basic component RepChain is selected as the underlying framework. RepChain is an alliance chain implemented by responsive programming, which has a good foundation in theory and engineering. Figure 7 shows the construction scheme of stakeholder alliance and consensus node.

- (1) RepChain is used as the evidence chain of medical waste data, and medical institutions, transportation departments, transportation companies, environmental protection departments, disposal companies, and other subjects are used as alliance nodes, whose cores are various devices with the capabilities of computing, data storage, and telecommunication. Regulatory agencies and operators can participate in the RepChain as a node using these connected

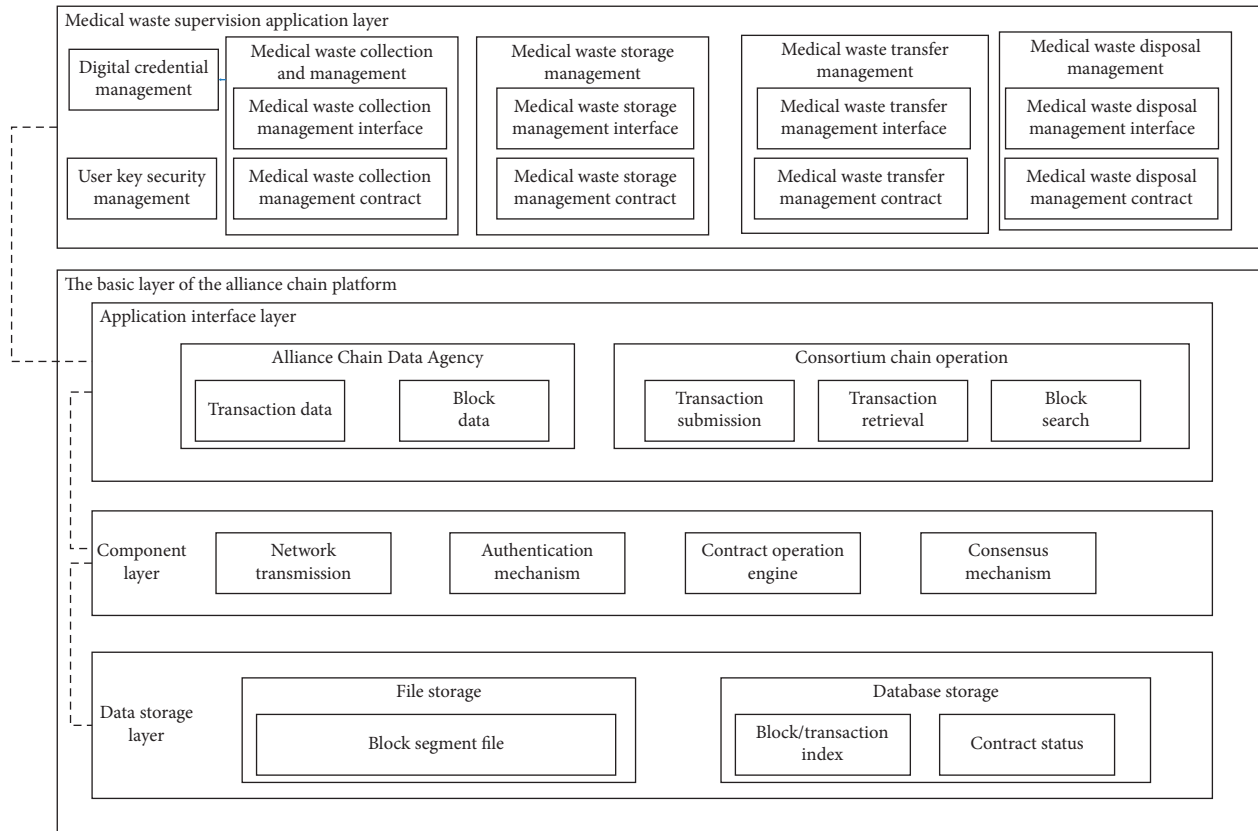


FIGURE 6: Regulatory framework for medical waste.

devices, and the critical data of key links are linked in the whole process to ensure the authenticity and integrity of data.

- (2) Smart contracts are used to store and retrieve medical waste and submit it to the blockchain in the form of signature transactions to ensure that all participants have clear rights and responsibilities.
- (3) The block information is viewed according to the visual real-time state diagram provided by RepChain. Any institution or user who can access the blockchain data can view the data on the chain and verify the authenticity of the information on the chain.

4.3. Deployment and Implementation of Medical Waste Supervision System. The deployment architecture of the medical waste supervision system is composed of a front exchange area, Web layer, application server layer, database server layer, and storage area. Our goal is to establish a service with quick, reliable, and sustainable responses. The structure is shown in Figure 8.

4.3.1. Front Exchange Layer. By deploying a set of front-end processors, it can receive data from outside, develop a unified interface and data standards, realize the function of data exchange, receive and send data files in real time or regularly, and separate the data exchange module from other

functional modules of the platform, which reduces the risk of operation and improves the reliability of the system.

4.3.2. Web Layer. According to the use of resources, the information request operation is forwarded, and the user's access platform is judged and switched according to the user's request. When Web browser connects to a server and requests a file, the server processes the request and sends the file to the Web browser, along with information that tells the browser how to view the file. The server uses the HTTPS protocol for information exchange to ensure the security of information transmission.

4.3.3. Application Server Layer. It mainly deploys the application program supporting the business implementation, specifically processes the request transmitted by the Web layer, and returns the corresponding processing results according to the corresponding business logic. The application server uses load balancing technology on multiple servers to bear the access pressure of the whole system according to certain rules. Each server has the same status, which can handle the high concurrent requests that one server cannot bear at the same time and cannot affect the operation of the whole system when one server fails.

4.3.4. Database Server Layer. It is mainly used to process data query or data manipulation requests, and the

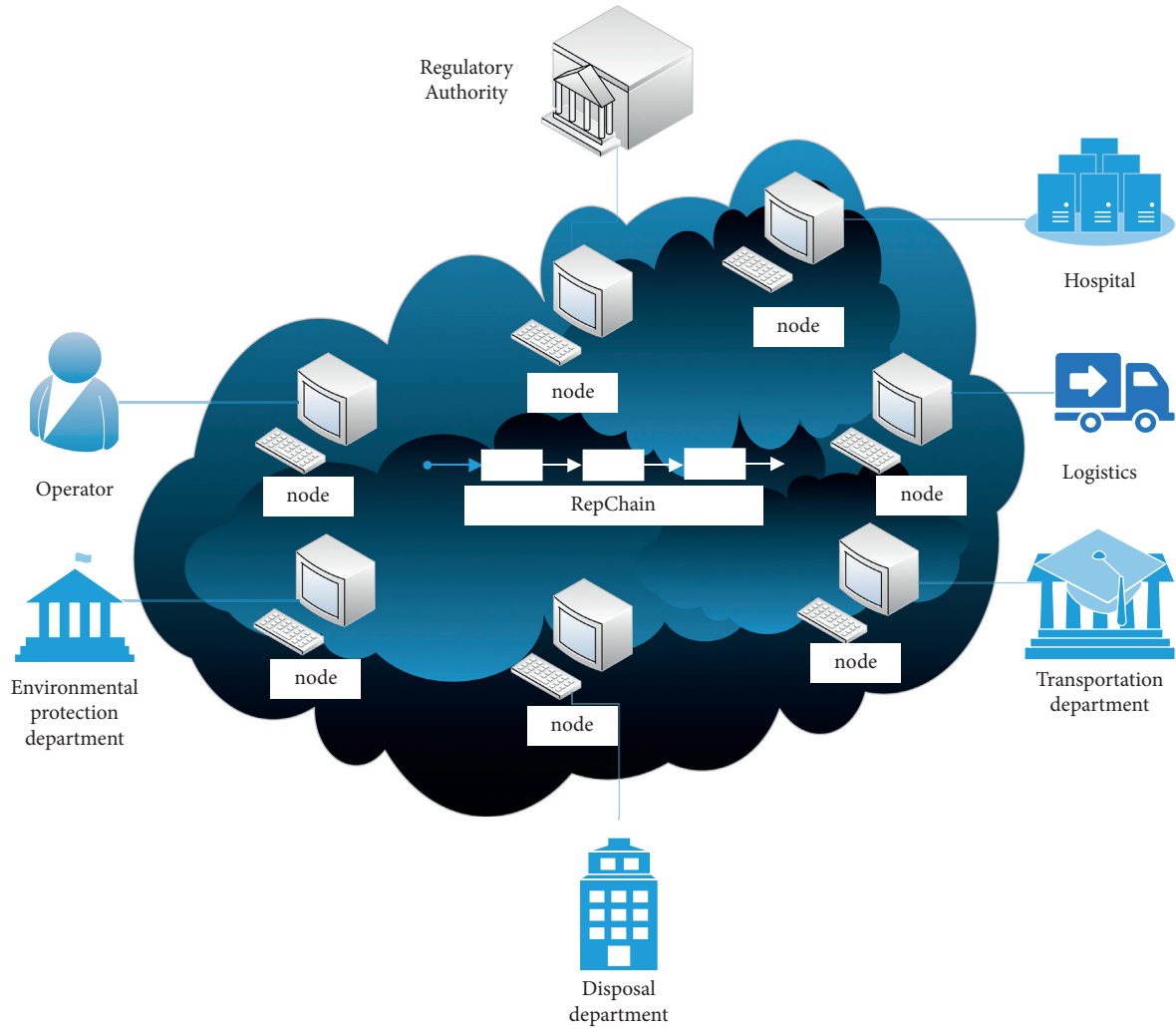


FIGURE 7: Stakeholder alliance and consensus node construction scheme.

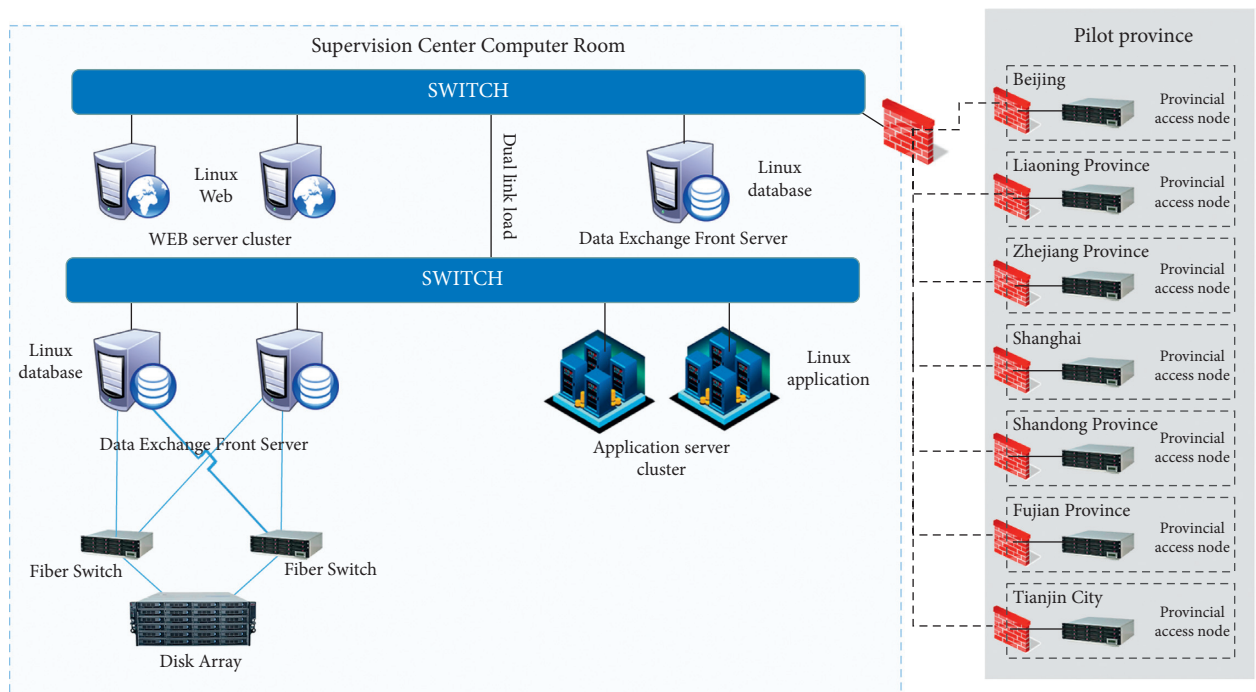


FIGURE 8: Deployment structure of medical waste supervision system.

application part that interacts with the user runs on the user's workstation. At the same time, transaction management, indexing, caching, query optimization, security, and multiuser access control are implemented.

4.3.5. Storage Area. It is mainly composed of a fiber switch and a disk array. The disk array comprises many cheaper disks, which are combined into a disk group with huge capacity. The performance of the whole disk system is improved by using the additive effect of the data provided by individual disks.

The format of medical waste data is designed, including fields such as name, type, and description. See Table 1 for details.

Uplink data is divided into two major categories: data oriented to account permissions and data oriented to business. The chaining data for account authority mainly includes the digital certificates of each participant for account authority management, wherein each participant can use the account management contract to establish an account for a natural person in charge of specific business activities as needed, and the digital certificates are linked in the same form.

Business-oriented uplink data can be divided into two categories: basic information data and full-process traceability information data. These two parts of information collected by nodes clarify the responsibility and content of each step and make further consolidation and arrangement possible. Among them, the basic information data mainly refers to the data of relative forms unrelated to the transfer and treatment of medical waste, including but not limited to the following:

- (i) Basic information on medical waste
- (ii) Basic information of the medical institution
- (iii) Basic information of the department
- (iv) Information of medical waste temporary storage point
- (v) Information on medical waste transfer vehicles
- (vi) Medical waste destruction point information
- (vii) Information on various types of personnel

The information of the input participant is linked to its digital certificate and identity. The whole-process traceability information data mainly refers to the dynamically increased data generated from the generation, classification, packaging, temporary storage, in-hospital transfer, out-of-hospital transfer, and final disposal of medical waste, including but not limited to the following:

- (i) Packaging traceability code
- (ii) Medical waste classification label
- (iii) Weighing information
- (iv) Documentary photos
- (v) Handover information
- (vi) Trajectory information during transportation

- (vii) Surveillance video evidence information

In the chaining process, the data related to a single participant needs to be signed by the participant with the private key corresponding to its digital certificate to ensure that the participant has an undeniable responsibility for the data. The data generated by the handover activity involving multiparty participation needs to be signed by the private key corresponding to the digital certificate of the multiparty participants and then can be linked to ensure that the participants agree that the data is authentic and credible. It is also necessary to establish the association relationship between the data records through unique identification and reference for the uplink information. In addition, each link forms a closed loop by recording the identifier of the preceding step in the chain-up data generated by the handover step, which means the procedures are well connected at an information level. The critical data can be stored in the latest chain-up state through a set structure such as an array, so that the state can be quickly viewed.

5. Demonstration of Medical Waste Supervision Model Based on Blockchain

The IoT-based and blockchain-based medical waste supervision model proposed in former sections has been employed in the industry and shown the capacity for efficient and reliable digitalized management of waste products, which is a valid proof of the feasibility of our system.

5.1. Practical Application of the Model. The research and development of the medical waste monitoring system were completed in 2018, and the system was put into operation in pilot hospitals in Shaanxi Province in June 2020. See Tables 2 and 3 for the cumulative monitoring of medical waste disposal. Table 2 presents the statistics of medical waste categories, namely, infectious waste, loss waste, pharmacological waste, chemical waste, and pathological waste; Table 3 shows the statistics of handover links, namely, department handover, nursing handover, warehouse management handover, and transportation handover.

The medical waste smart contract deposit algorithm is shown in Figure 9, and the traceability algorithm is shown in Figure 10.

Figure 9 introduces the storage certificate algorithm. The blockchain-based medical waste disposal system uses the storage certificate smart contract to upload medical waste-related information to the blockchain, which mainly includes basic medical waste information, basic information of medical institutions, basic information of departments, and medical treatment. The information is mainly collected from temporary waste storage sites, medical waste transfer vehicles, medical waste destruction sites, various personnel information, and so on. Each user has an independent private key to ensure that only users or institutions permitted by the blockchain can initiate requests. If an unregistered user or institution initiates the request, it will be directly rejected. Regarding the data on the chain, a hash function is used to encrypt the string to generate a unique, conflict-free, and

TABLE 1: Categories of medical wastes.

Name	Type	Description	Remark
PackageId	String	Waste bag number	The smallest unit of waste
BoxId	String	Shipping box number	Packing the waste and packing it into a box
VehicleId	String	Transport vehicle number	Loading waste after packing
MW_Type	String	Waste category	Plaintext
MW_Weight	String	Waste weight	
Op_Role	String	On-chain role category	(1) Hospital (2) Transportation (3) Disposal
Op_UnitId	String	Operator unit ID	
Op_MemberId	String	Operator ID	
Op_MemberDigest	String	Operator information digest hash	(1) Initial packaging/confirmation (2) Packing/confirmation (3) Loading/confirmation (4) Temporary storage/confirmation of warehousing (5) Recycling/confirmation
Operation	String	Operational behavior	
Oth_MemberId	String	Handover ID	
Op_DateTime	String	Operating time	Timestamp
Op_Location	String	Operating location coordinates	Traditional latitude and longitude format
Op_RFIDInfo	String	RFID location coordinate information	
Memo_Photo	String	Photo digest hash	

Note. The above table only shows the design of some uplink data formats.

TABLE 2: Statistics of categories of medical wastes.

Medical waste category	Weight (kg)
Infectious waste	1467.84
Lossy waste	105.58
Pharmacological waste	81.32
Chemical waste	43.05
Pathological waste	113.68

TABLE 3: Statistics of handover link.

Handover	Quantity (article)
Department handover	889
Nurse handover	88
Library management handover	54
Transport handover	49

irreversible identifier. The PROOF function is used to store medical waste-related information on the blockchain. Before that, it will determine whether there is duplicate data on the blockchain. If there is, it will show that it has been on the chain. If not, it will be stored on the blockchain.

Figure 10 introduces the traceability algorithm. The retrieval function user checks whether the comparison chain is consistent between the upper and lower chains. After the medical waste-related data is stored in the blockchain, when the verifier wants to verify whether the data is true and reliable, on the one hand, it can pass any section. The three-party tool performs verification, submission, and production of electronic evidence document summaries. On the other hand, it can obtain the deposit transaction from the block of the blockchain and obtain the data set from it, so that the two can be compared and the verification result can be obtained from it.

5.2. Application Effect Demonstration. The medical waste supervision system saves the tedious manual recording and data analysis for medical institutions and realizes the excellent management of medical waste by collecting required information and facilitating data exchange. Similarly, only smartphones need to be used for health supervision agencies to trace back and monitor the whole process of medical waste disposal in real time, around the clock. The medical waste supervision system interface is shown in Figure 11, and Figure 12 shows the signature and verification of the blockchain transaction data.

Through the innovative blockchain application and the IoT, the medical waste supervision system promotes the standardized and digitalized management of medical waste. Through online real-time monitoring, whole-process monitoring, and other informatized means, it focuses on solving the difficulties of medical waste supervision in health supervision and improves the informatization and intelligence level of health supervision.

5.3. Related Research. The studies listed in Table 4 are more relevant to this article. Compared with these works, the advancement of this work is reflected in privacy protection. We propose a verifiable credential implementation method based on atomic signature. The method includes the following: (a) a credential structure creator defines and creates a specific verifiable credential structure with various properties. (b) Based on the atomic signature mechanism, the credential issuer constructs a verifiable credential complying with the above verifiable credential structure. (c) The credential holder checks the validity of the verifiable credential. (d) The credential holder selectively discloses and presents the specific attributes information with the

Algorithm 1: PROOF (ctx, data), Medical waste deposit certificate deposit function module in smart contract

Input: ctx, ContractContext, api: Shim, t: Transaction.
 data, Important elements of medical waste, map [string, string].
 data= proofType, package, id, proofDate, mW_Weight, op_Datetime, op_Location, operation...

Output: v, the identity of the medical waste deposited successfully or null.

```

1 repchain_cert ← getCert (t.initiator_id)
2 if repchain_cert and verify (repchain_cert.pubkey, t, t.sig)
3   and repchain.cert complies with X.509 standards
4   and verify (t.cert.pubkey, t.cert, t.cert.sig) then
5     k ← get (file_hash)
6     pv0 ← ctx.api.getVal (k)
7     if pv0 ∉ null then
8       return pv0
9     else
10      putProof ← ctx.api.setVal (k,v)
11      return putProof
12   end if
13 else
14   return null
15 end if

```

FIGURE 9: Medical waste smart contract certificate algorithm.

Algorithm 2: Retrieval

Input: ctx, ContractContext, api: Shim, t: Transaction.
 data, Important elements of medical waste, map [string, string].
 data= proofType, package, id, operation...

Output: v, the identity of the medical waste deposited successfully or null.

```

1 repchain_cert ← getCert (t.initiator_id)
2 if repchain_cert and verify (repchain_cert.pubkey, t, t.sig)
3   and repchain.cert complies with X.509 standards
4   and verify (t.cert.pubkey, t.cert, t.cert.sig) then
5     val proofIndex = index.proofType + SPLIT_CHAR + index.id
6     val value = ctx.api.getVal (proofIndex) .asInstanceOf [String]
7     if (value == null)
8       "nothing be retrieved"
9     else
10      value
13  else
14    return null

```

FIGURE 10: Intelligent contract traceability algorithm for medical waste.

respective atomic signatures to a credential verifier. (e) When receiving the presentation, the credential verifier verifies its authenticity and validity. (f) The credential issuer could either entirely update the whole verifiable credential or partly update some attributes of the verifiable credential.

5.4. Case Study. This work has been successfully applied in Shaanxi Province, China, in 2020, and because of certain restrictions, we cannot get specific data. Therefore, we have

added a case study to the article and conducted some analysis on the reasons for the successful application of this work in Shaanxi Province.

In 2019, the National Health Commission of China randomly inspected 41,337 medical and health institutions for medical waste and imposed administrative penalties on 2,122 institutions that violated the relevant regulations on medical waste disposal, accounting for 80% of the number of investigations and punishments of medical and health institutions for violations of infectious disease prevention and control.

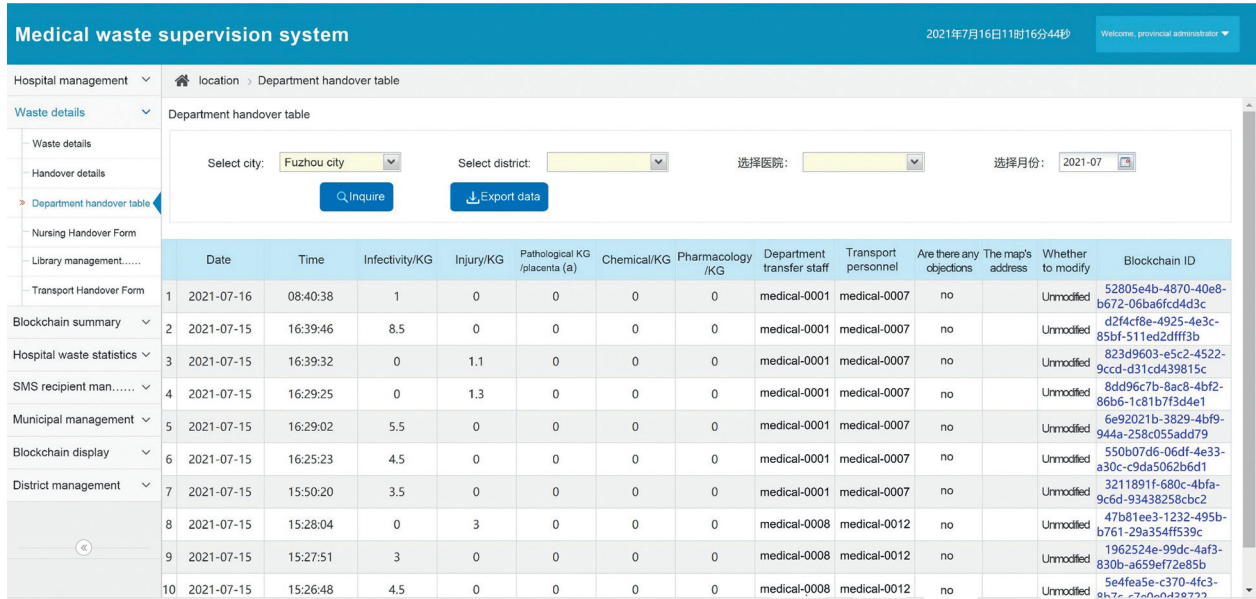


FIGURE 11: Interface of medical waste supervision system.

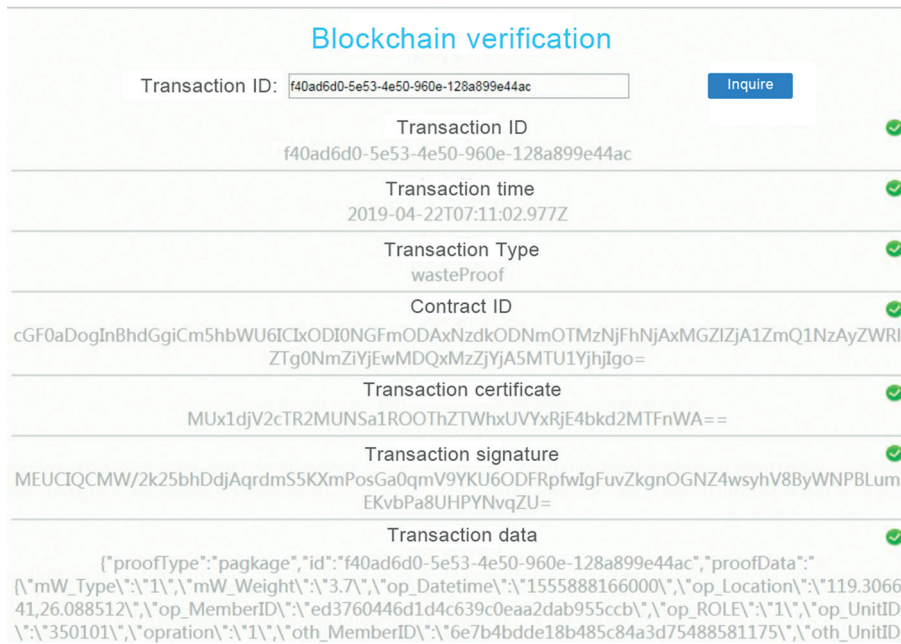


FIGURE 12: Signature and verification of blockchain transaction data.

TABLE 4: Statistics of categories of medical wastes.

Study	Waste type	Shipping rules	Traceability	Decentralized	Supply chain	Waste frauds	Privacy protection
[30]	Domestic waste	No	Yes	No	No	No	No
[31]	Electronic waste	No	Yes	No	Forward	No	No
[32]	Domestic waste	No	No	No	No	No	No
[33]	N/A	No	Yes	Yes	Forward	No	No
[29]	Domestic waste	No	No	No	No	No	No
[24]	COVID-19 MW	Yes	Yes	Yes	Forward	Yes	No
Our study	Medical waste	Yes	Yes	Yes	Forward	Yes	Yes

In recent years, the administrative departments of health and environmental protection, through deepening the reform of the medical and health system, are focusing on solving the difficulties and pain points in managing medical waste disposal in small and medium medical institutions. Examples include (a) the integrated management of rural medical and health institutions; (b) the functions of new medical management mechanisms such as the medical community; (c) exploring the management model of centralized medical waste from primary medical and health institutions to the higher-level medical and health institutions for unified disposal; and (d) transporting to the nearest medical waste centralized disposal unit that holds a hazardous waste business license, namely, suitable disposal.

This work explores and promotes the implementation of the “blockchain + medical waste supervision” model in Shaanxi Province, China. We use blockchain technology to digitally monitor the entire process of medical waste generation, storage, and transfer in medical and health institutions. The traces of medical waste can be detected throughout the entire process. Through big data analysis, we can grasp the actual production and centralized disposal weight of various types of medical waste and respond to abnormal situations in a timely manner.

In May 2021, at the National Medical Waste Management Work Conference held in Shaanxi Province, the Shaanxi Provincial Health Commission summarized and promoted the “blockchain + medical waste supervision” model to all provinces. In this way, the role of blockchain technology in supervision during and after the event has been fully brought into play. Blockchain technology promotes regulatory innovation, promotes the modernization of medical waste management capabilities, and realizes the maximization of regulatory efficiency, the optimization of regulatory costs, and the minimization of human interference.

6. Conclusion

This paper proposes a medical waste supervision model based on blockchain, which combines blockchain technology with digital evidence provided by directly involved individuals or institutions. On the one hand, it can provide an efficient and transparent way of supervision, ensure the authenticity and integrity of data, improve the medical waste supervision system, and enhance the credibility of regulations. On the other hand, the detailed information of personnel is encapsulated in digital credentials, which effectively solves the user privacy problem existing in the traditional medical waste supervision processes and can effectively promote the healthy development of the medical waste supervision industry. The decentralized IoT model for medical waste supervision proposed in this paper provides a new idea for the transformation and upgrading of existing medical waste supervision models and implementation methods. Furthermore, the model shows the capability and importance of IoT-based blockchain in integrating and managing dispersed information which can bring tangible and revolutionary changes to various areas.

The possible future directions of this work can be the application of the proposed model in different scenarios. In addition, other algorithms and technologies can be considered to further improve the performance and portability of the system.

Data Availability

All the data are collected in the medical waste monitoring system since our paper focuses on building this system rather than investing in the data. If anyone pursues further investigation of the available data, please contact the corresponding author. Moreover, we will help in extracting the data from the real-world system.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

- [1] M. Kassou, S. Bourekkadi, S. Khouli, K. Slimani, H. Chikri, and M. L. Kerkeb, “Blockchain-based medical and water waste management conception,” in *Proceedings of the International Conference on Innovation, Modern Applied Science & Environmental Studies (ICIES2020)*, Kenitra, Morocco, February 2021.
- [2] A. S. L. França, J. A. Neto, R. F. Gonçalves, and C. M. V. B. Almeida, “Proposing the use of blockchain to improve the solid waste management in small municipalities,” *Journal of Cleaner Production*, vol. 244, Article ID 118529, 2020.
- [3] A. Kalla, T. Hewa, R. A. Mishra, M. Ylianttila, and M. Liyanage, “The role of blockchain to fight against COVID-19,” *IEEE Engineering Management Review*, vol. 48, no. 3, pp. 85–96, 2020.
- [4] A. A. Abd-Alrazaq, M. Alajlani, and D. Alhuwail, “Blockchain technologies to mitigate COVID-19 challenges: a scoping review,” *Computer Methods and Programs in Biomedicine Update*, vol. 1, Article ID 100001, 2020.
- [5] M. Kouhizadeh, S. Saberi, and J. Sarkis, “Blockchain technology and the sustainable supply chain: theoretically exploring adoption barriers,” *International Journal of Production Economics*, vol. 231, Article ID 107831, 2021.
- [6] Z. Zheng, S. Xie, H. N. Dai, X. Chen, and H. Wang, “Blockchain challenges and opportunities: a survey,” *International Journal of Web and Grid Services*, vol. 14, no. 4, pp. 352–375, 2018.
- [7] D. Berdik, S. Otoum, N. Schmidt, D. Porter, and Y. Jararweh, “A survey on blockchain for information systems management and security,” *Information Processing & Management*, vol. 58, no. 1, Article ID 102397, 2021.
- [8] S. Tanwar, K. Parekh, and R. Evans, “Blockchain-based electronic healthcare record system for healthcare 4.0 applications,” *Journal of Information Security and Applications*, vol. 50, Article ID 102407, 2020.
- [9] H. Feng, X. Wang, Y. Duan, J. Zhang, and X. Zhang, “Applying blockchain technology to improve agri-food traceability: a review of development methods, benefits and challenges,” *Journal of Cleaner Production*, vol. 260, Article ID 121031, 2020.
- [10] L. Tseng, L. Wong, S. Otoum, M. Aloqaily, and J. B. Othman, “Blockchain for managing heterogeneous internet of Things: a

- perspective architecture,” *IEEE Network*, vol. 34, no. 1, pp. 16–23, 2020.
- [11] R. W. Ahmad, K. Salah, R. Jayaraman, and I. Yaqoop, “Blockchain-based forward supply chain and waste management for COVID-19 medical equipment and supplies,” *IEEE Access*, vol. 9, pp. 44905–44927, 2021.
- [12] X. Li, P. Jiang, T. Chen, X. Luo, and Q. Wen, “A survey on the security of blockchain systems,” *Future Generation Computer Systems*, vol. 107, pp. 841–853, 2020.
- [13] F. Jamil, S. Ahmad, N. Iqbal, and D.-H. Kim, “Towards a remote monitoring of patient vital signs based on IoT-based blockchain integrity management platforms in smart hospitals,” *Sensors*, vol. 20, no. 8, 2195 pages, 2020.
- [14] P. Singh and N. Singh, “Blockchain with IoT and AI,” *International Journal of Applied Evolutionary Computation*, vol. 11, no. 4, pp. 13–27, 2020.
- [15] S. B. Rane and S. V. Thakker, “Green procurement process model based on blockchain–IoT integrated architecture for a sustainable business,” *Management of Environmental Quality: International Journal*, vol. 31, no. 3, pp. 741–763, 2019.
- [16] S. E. Chang, Y.-C. Chen, and M.-F. Lu, “Supply chain re-engineering using blockchain technology: a case of smart contract based tracking process,” *Technological Forecasting and Social Change*, vol. 144, pp. 1–11, 2019.
- [17] C. R. Bass, B. Benefield, D. Horn, and R. Morones, “Increasing robustness in long text classifications using background corpus knowledge for token selection,” *SMU Data Science Review*, vol. 2, no. 3, 10 pages, 2019.
- [18] C. X. Li, S. Chen, L. S. Zheng, C. Zuo, B. Y. Jiang, and G. Liang, “RepChain—a permissioned blockchain toolkit implemented by reactive programming,” *Journal of Software*, vol. 30, no. 6, pp. 1670–1680, 2019.
- [19] Q. Xia, W. S. Dou, K. W. Guo, G. Liang, C. Zuo, and F. J. Zhang, “Survey on blockchain consensus protocol,” *Journal of Software*, vol. 32, no. 2, pp. 277–299, 2021.
- [20] N. Singh, Y. Tang, Z. Zhang, and C. Zheng, “COVID-19 waste management: effective and successful measures in Wuhan, China,” *Resources, Conservation and Recycling*, vol. 163, Article ID 105071, 2020.
- [21] H. Liu and Z. Yao, “Research on the reverse logistics management of medical waste based on the RFID technology,” *Fresenius Environmental Bulletin*, vol. 26, pp. 8084–8092, 2017.
- [22] K. Wang and W. Nai, “Application of 5G wireless communication technology in hazardous medical waste treatment,” in *Proceedings of the 2021 IEEE international conference on software engineering and artificial intelligence (SEAI)*, pp. 87–90, IEEE, Xiamen, China, June 2021.
- [23] S. Li, Y. Lin, R. Ma, and Q. Chen, “Research on the integrated development of “blockchain + medical waste treatment”,” *Technology and Innovation*, vol. 4, no. 17, pp. 20-21+24, 2019.
- [24] R. W. Ahmad, K. Salah, R. Jayaraman, I. Yaqoob, M. Omar, and S. Ellahham, “Blockchain-based forward supply chain and waste management for COVID-19 medical equipment and supplies,” *IEEE Access*, vol. 9, pp. 44905–44927, 2021.
- [25] X. Jiang and Q. Tian, *A Blockchain-Based Medical Waste Supervision Platform and Management Method*, CN112259203A, Beijing, China, 2020.
- [26] Z. Lin, *On-site Treatment and Supervision System of Medical Waste Based on Internet of Things and blockchain*, CN112990494A, Beijing, China, 2021.
- [27] N. Gupta and P. Bedi, “E-waste management using blockchain-based smart contracts,” in *Proceedings of the 2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, Chengdu, China, June 2018.
- [28] M. R. Laouar, Z. T. Hamad, and S. Eom, “Towards blockchain-based urban planning: application for waste collection management,” in *Proceedings of the 9th International Conference on Information Systems and Technologies (icist 2019)*, Association for Computing Machinery, New York, NY, USA, March 2019.
- [29] J. Stark, “Product lifecycle management,” *Product Lifecycle Management*, vol. 2, pp. 1–35, 2016.
- [30] M. Lamichhane, “A smart waste management system using IoT and blockchain technology,” Master’s Thesis in PERCOM Master Program, ITMO University, Saint Petersburg, Russia, 2017.
- [31] N. Gupta and P. Bedi, “E-waste management using blockchain based smart contracts,” in *Proceedings of the International Conference Advanced Computer Communication Information (ICACCI)*, pp. 915–921, Bangalore, India, September 2018.
- [32] G. K. Shyam, S. S. Manvi, and P. Bharti, “Smart waste management using Internet-of-Things (IoT),” in *Proceedings of the 2nd International Conference Computer Communication Technology (ICCCCT)*, pp. 199–203, Chennai, India, February 2017.
- [33] I. Omar, M. Debe, R. Jayaraman, K. Salah, M. Omar, and J. Arshad, “Blockchain-based supply chain traceability for COVID-19 PPE,” *Computers & Industrial Engineering*, vol. 167, Article ID 107995, 2022.

Research Article

Lightweight and Anonymous Mutual Authentication Protocol for Edge IoT Nodes with Physical Unclonable Function

Hongyuan Wang,¹ Jin Meng,² Xilong Du,¹ Tengfei Cao,² and Yong Xie ²

¹Qinghai Province Yindajihuang Project Construction and Operation Bureau, Xining, China

²Department of Computer Technology and Application, Qinghai University, Xining, China

Correspondence should be addressed to Yong Xie; mark.y.xie@qq.com

Received 10 September 2021; Accepted 23 October 2021; Published 4 January 2022

Academic Editor: Jie Cui

Copyright © 2022 Hongyuan Wang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Internet of Things (IoT) has been widely used in many fields, bringing great convenience to people's traditional work and life. IoT generates tremendous amounts of data at the edge of network. However, the security of data transmission is facing severe challenges. In particular, edge IoT nodes cannot run complex encryption operations due to their limited computing and storage resources. Therefore, edge IoT nodes are more susceptible to various security attacks. To this end, a lightweight mutual authentication and key agreement protocol is proposed to achieve the security of IoT nodes' communication. The protocol uses the reverse fuzzy extractor to acclimatize to the noisy environment and introduces the supplementary subprotocol to enhance resistance to the desynchronization attack. It uses only lightweight cryptographic operations, such as hash function, XORs, and PUF. It only stores one pseudo-identity. The protocol is proven to be secure by rigid security analysis based on improved BAN logic. Performance analysis shows the proposed protocol has more comprehensive functions and incurs lower computation and communication cost when compared with similar protocols.

1. Introduction

With the rapid development of new network technologies such as cloud computing and artificial intelligence, Internet of Things (IoT) has been more and more widely used. It has continuously brought great convenience to people's lives and work [1]. IoT devices play an important role in the power generation, transmission, and distribution of smart grids and can monitor power transmission conditions in a more timely manner [2]. A system called iERS can monitor and notify the availability of parking spaces near the smart community through the IoT infrastructure and help users find suitable parking spaces [3]. Baker et al. [4] created a general model that can be used in most similar healthcare systems using end-to-end IoT. Therefore, diverse technologies based on the IoT make users' comfortable and convenient life possible.

According to the predictions of relevant agencies, IoT devices are expected to grow exponentially in the next few

years, followed by the explosive growth of IoT data [5]. In some low-latency IoT applications, the design idea of combining the computing functions of the edge cloud to complete the reception and management of massive data has become a way to improve the efficiency of IoT. Edge cloud helps edge IoT nodes process data nearby, reducing the heavy computing tasks of cloud data centers.

However, due to the openness of channels and data sensitivity, data security and user privacy issues have attracted more and more attention. Data security issues are also one of the biggest obstacles restricting the widespread deployment and application of Internet of Things [6]. Due to IoT characteristics, the specific challenges faced by data security are as follows: (1) IoT device resources are generally limited. Internet of Things consists of many heterogeneous and resource-constrained devices, which often have a single function and limited computing and storage resources [7]; (2) massive data: the number of IoT devices and users is huge, and massive amounts of data are generated in real

time, which brings great workload to security authentication; (3) interactive dynamics: in the environment of Internet of Things, nodes and users are often in constant movement, which makes real-time requirements for secure access and authentication; and (4) strong data privacy: the advent of the big data era puts forward higher requirements for the protection of personal privacy information, and both visitors and IoT nodes must be protected [8].

In order to solve the above-mentioned IoT data security issues, many researchers have proposed various security authentication and key agreement protocols to solve the IoT data security issues [9]. However, as we all know, Internet of Things has many remote nodes. In this scenario, an attacker can extract stored authentication information and keys from the IoT device and then can perform security attacks according to their own needs. At present, most studies have not considered this aspect of security issues. Therefore, the communication protocol designed for the IoT system should ensure that the entire system remains secure, even if the equipment or sensors are damaged. Fortunately, physical unclonable functions (PUF) provide a viable option to achieve this goal. Recently, some PUF-based authentication protocols have been proposed to protect sensor security and data security.

To solve the above issues, we propose a lightweight and anonymous mutual authentication protocol for edge IoT nodes with physical unclonable function. The proposed protocol only needs some lightweight cryptographic operations and stores one pseudo-identity. It is very suitable for data security protection scenarios of IoT nodes in a wide range of deployment scenarios. To sum it up, the main contributions of the proposed protocol are as follows:

- (i) The proposed protocol realizes secure, lightweight mutual authentication for edge IoT nodes. More importantly, in addition to the noise of the nonideal PUF, we also take the imbalance of resources between the device and the server into account, taking advantage of the reverse fuzzy extractor to reduce the cost.
- (ii) The proposed protocol only store one pseudo-identity to prevent physical security attack such as side-channel security attacks and memory data theft while ensuring anonymity.
- (iii) We introduced a supplementary subprotocol for desynchronization attacks to overcome the shortcomings in [10]. It also improves efficiency by querying the relevant subset in the database based on the registration time instead of traversing the entire subset.
- (iv) We present rigid security proof based on improved BAN logic [11] to demonstrate the proposed protocol is against all of secure attacks.

The paper's organization is as follows: Section 2 shows the related works on the authentication protocols for the IoT system. Section 3 and Section 4 introduce, respectively, related preliminaries and system model and security requirements. Section 5 presents the proposed scheme with its

supplementary subprotocol in detail. Section 6 and Section 7 show the security and performance analysis. Finally, the conclusion and future work are described in Section 8.

2. Related Works

As IoT has gained steam in recent decades, its security issues have also attracted widespread attention. In 2014, a study by Hewlett Packard suggested that about seventy percent of IoT devices suffer from acute vulnerability, which cannot be ignored [12]. Therefore, considerable authentication protocols for Internet of Things sprang up.

Most of the incipient authentication protocols are based on asymmetric cryptography, which cuts both ways in IoT: it boasts higher security but bears inevitably the computational inefficiency and huge overhead. For instance, Fouda et al. [13] proposed a scheme that established the shared session key with Diffie–Hellman exchange protocol, whose needed computing resources put a certain burden on resource-constrained IoT devices. In addition, Porambage et al. [14] involved the elliptic curve cryptography belonging to the public key system to achieve the implicit certificate-based protocol. Besides, Amin et al. [15] utilized the smart card and the RSA algorithm. Therefore, not only does it have a major potential danger in tampering because it is vulnerable to physical attack but also it contributes to terribly large computation costs.

Then, the study on protocols with symmetric cryptography is generally extensive. Das et al. [16] introduced a scheme with smart cards, which is a novel authentication protocol on the basis of passwords and symmetric cryptography for the hierarchical wireless sensor networks (HWSN), a branch of Internet of Things. However, it is similar that the scheme, which is not tamper-proof, cannot avoid physical attacks. Turkanovi and Holbl [17] designed another protocol for HWSN, which pointed out the flaws in [16] and eliminated its redundant components, taking advantage of the symmetric encryption or decryption. Nevertheless, even if symmetric cryptography reduces the computational complexity and saves some resources with hash functions, XOR operations, and concatenation operations, compared with the asymmetric one, the storage of secret keys still produces a large memory overhead in a matter of the IoT system connected with a substantial amount of devices.

The demand for more secure and efficient authentication protocols has prompted scholars to introduce the PUF, which makes up for the drawbacks of smart cards and is claimed as a hardware function with great promise in recent research. Aman et al. [18] showed the scheme where the response generated by PUF encrypted the data and verified the source. Chatterjee et al. [19] proposed the scheme which used the response value to construct the session key. What is more, there is no need to explicitly store the challenge-response pair. However, the protocols mentioned in [18, 19] fail to guarantee anonymity. In addition, the challenge-response pair is not updated and replaced every round, even when the protocol introduced by Feikken et al. [20] avoids conveying the identity in plain text. Consequently,

considering the device anonymity, Gope and Sikdar [10] presented a scheme with plentiful alternative pseudonyms and challenge-response pairs. Instead of direct identity, it completes communication with the help of pseudo-identity which, together with the challenge-response pair, is regenerated to prevent adversaries from the trail. However, it is more likely to encounter desynchronization attacks. The protocol proposed by Jiang et al. [21] resolved the above two weaknesses, but its overhead increases due to asymmetric cryptography. Additionally, the protocol in [22] performs better than that in [10] in terms of resistance to desynchronization attack. On the contrary, the majority of protocols such as [18] merely consider the ideal PUF. Since noisy factors are inescapable in daily life, it is required to take appropriate measures against them. Significantly, the fuzzy extractor is regarded as a widely used and practical tool for error correction. In the part of noisy PUF in [22], the fuzzy extractor emerges to convert the error response values. Besides, the protocol in [20] also serves as an example to show the great role of the fuzzy extractor in addressing noisy PUF issues. Furthermore, the fuzzy extractor in reverse is a feasible optimization method, which takes the resource difference between the device and the server in IoT system into full consideration and makes the resource utilization more reasonable. For instance, the protocols in [10, 21, 23, 24] reverse the fuzzy extractor to arrange resources more evenly.

3. Preliminaries

3.1. Physical Unclonable Function. Described as “an expression of an inherent and unclonable instance-specific feature of a physical object” in [25], the PUF is considered a key factor in the physical uniqueness of a device. Thanks to the randomness and uncertainty during the fabrication of integrated circuits, it is less likely to produce a copy; thereby, the PUF is increasingly shining in the security domain.

Additionally, the definition in [26] that a PUF is deemed to be a special function that inputs a random challenge and generates the corresponding response relying on the complex physical character clarifies the PUF from another perspective. As shown in the following equation, C is the challenge inputted and R is the response outputted:

$$R = \text{PUF}(C). \quad (1)$$

In ideal circumstances, there is a one-to-one correspondence between the challenge-response pair and the PUF; scilicet, if a challenge is assigned to the same PUF multiple times, the responses generated are identical, and if the same challenge is given to different PUFs, the responses obtained are distinct. However, due to the environmental and circuit noise, a PUF always outputs various responses with a few errors to a challenge value.

3.2. Reverse Fuzzy Extractor. Since the influence of noisy PUFs cannot be ignored, the fuzzy extractor is introduced to address the issue. Combined with the PUF, the fuzzy

extractor with a secure sketch maps the responses with resemblance to the same result [27].

A fuzzy extractor (m, l, t, ε) comprises two algorithms, which are $\text{Gen}(\cdot)$ and $\text{Rec}(\cdot)$, according to [20,27]. As a probabilistic algorithm, $\text{Gen}(\cdot)$ generates a key string $k \in \{0, 1\}^l$ and a helper data hd with the input value R . In the phase, in terms of every R with min-entropy m , with (2), the difference of statistics between (k, hd) and (U_l, k) is up to the threshold ε . U_l means a constellation of strings from $\{0, 1\}^l$, which are chosen in a random and uniform way. As a deterministic algorithm, if the hamming distance between R and R' is at most t , $\text{Rec}(\cdot)$ can utilize hd and R' to reproduce k , according to (3):

$$(k, hd) = \text{Gen}(R), \quad (2)$$

$$k = \text{Rec}(R', hd). \quad (3)$$

Generally, the reconstruction function $\text{Rec}(\cdot)$ is deployed on the device with a PUF, while the key generation function $\text{Gen}(\cdot)$ is placed in the server. However, it is a critical defect that the reconstruction algorithm is performed on the device end with limited memory and computing resources as a consequence of numerous gates and time costs when correcting errors [28]. Therefore, the reverse fuzzy extractor, which sets $\text{Gen}(\cdot)$ on the PUF-equipped device and $\text{Rec}(\cdot)$ on the server, is applied to resolve the problem.

3.3. Symbols and Descriptions. The symbols and descriptions involved in the protocol are presented in Table 1.

4. System Model and Security Requirements

4.1. System Model. Figure 1 shows two roles in the system model: a series of IoT devices and a server situated in the data center. Moreover, the communication between devices and the server is through Internet in the IoT system.

- (i) IoT devices: In the IoT system, every device possesses a PUF, in which any effort to manipulate the PUF will make it unavailable and any attempt to remove the PUF will comprise it. In addition, it is assumed that devices have finite resources.
- (ii) Server: The server is described as a secure, trusted, and resource-unlimited entity, which can store the related information about IoT devices in the database to operate the mutual authentication.

4.2. Adversary Model. In matters of the adversary model, we refer to the well-known Dolev–Yao attack model in [29], with an assumption that an adversary A boasts a series of capabilities as described below:

- (i) According to the Dolev–Yao model, the adversary A has complete control over the open channel, who can grasp total information on the insecure channel between the IoT device D_i and the server S and thereby intercept, tamper, or cancel it.

TABLE 1: Symbols and descriptions.

Symbols	Descriptions
D_i	The identity of the IoT device
TD_i	The one-time temporary identity of IoT device
RT_i	The registration time
T_i	The current timestamp
(C_i, R_i)	The challenge-response pair
(N_i)	The nonce generated by the IoT device
(N_s)	The nonce generated by the server
sk	The session key
PUF	The physical unclonable function
Gen(.)	The key generation algorithm of the fuzzy extractor
Rec(.)	The reconstruction algorithm of the fuzzy extractor
$h(\cdot)$	The secure one-way hash function
\parallel	The concatenation operation
\oplus	The XOR operation

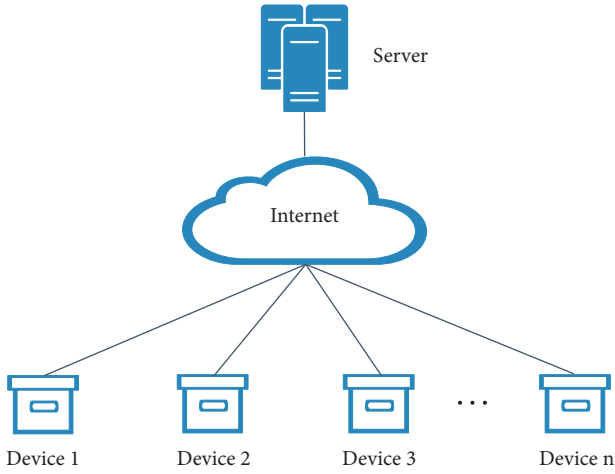


FIGURE 1: The system model.

- (ii) Besides the threats mentioned above, aiming at acquiring the essential data, the adversary can also launch physical attacks, cloning attacks, counterfeit attacks, desynchronization attacks, and so forth.

4.3. Security Requirements. After the analysis of the adversary model, we take account of the related security requirements for the proposed two-party authentication protocol:

- (i) Mutual authentication: The genesis of the fact that it is crucial to achieve the mutual authentication between the IoT device and the server before the formal communication lurks in the issue that an attacker may disguise as a trusted device sending malicious information to others with the impersonation attack.
- (ii) Reliable session key generation: The problem that an adversary is more likely to obtain the messages transmitted through the open channel serves as an explanation of the requirement that both the device end and the server end ensure the same session key is held during communication.

- (iii) Anonymity: It is indispensable to use one-time aliases so that the adversary cannot know the true identity of the device.
- (iv) Defense against the known attacks: The designed protocol is supposed to resist the known attacks, such as physical attacks, cloning attacks, impersonation attacks, and especially desynchronization attacks.

5. The Proposed Scheme

In this section, we propose a lightweight and anonymous mutual authentication protocol for edge IoT nodes with physical unclonable functions, which features the zero storage of shared secrets and a large number of pseudonyms. In total, the protocol is composed of three phases: the setup phase, the registration phase, and the authentication phase.

5.1. Setup Phase. In this stage, a reliable one-way hash function $h: (0, 1)^* \rightarrow \{0, 1\}^l$ is selected to achieve mutual authentication, where l is a secure parameter chosen by the server.

5.2. Registration Phase. In this stage, the IoT device sends its relevant messages to the server through the secure channel as shown in Figure 2. The IoT device selects a registration time RT_i (a time slot such as three days or five days), which together with the identity D_i is utilized to calculate $FR_i = \text{PUF}(D_i \parallel RT_i)$ in order to prepare for the supplementary subprotocol against the desynchronization attack. Then, the device randomly chooses a one-time temporary alias $TD_i \in \{0, 1\}^l$ and a challenge value $C_i \in \{0, 1\}^l$ and obtains the response R_i from the PUF. The device stores the TD_i needed in this round temporarily, while the registration time RT_i is also stored in a secure environment. Next, $\text{Msg}_0: \{D_i, TD_i, (C_i, R_i), FR_i, RT_i\}_i$ is sent to the server through the ideal channel. After receiving Msg_0 , the server stores it in the database.

5.3. Authentication Phase. In this stage, the device and the server in the IoT system conduct mutual authentication where a few pseudo-identities and shared secrets are stored by the device end. The final generation of the same session key on the device and the server means the achievement of their mutual authentication.

- (1) The IoT device transmits TD_i of this round to the server S . On receiving the alias, the server searches for it in the database. If found successfully, S gets the corresponding challenge-response pair (C_i, R_i) and selects a nonce N_s . Then, the server computes $N_s^* = h(D_i \parallel C_i) \oplus N_s$ and $h_s = h(N_s^* \parallel C_i)$. Finally, $\text{Msg}_1: \{C_i, N_s^*, h_s\}$ is given to the IoT device.
- (2) Upon receiving Msg_1 , the IoT device calculates $R'_i = \text{PUF}(C_i)$, $(k'_i, hd'_i) = \text{Gen}(R'_i)$, $N_s^* = h(D_i \parallel C_i) \oplus N_s^*$, and $h'_s = h(N_s^* \parallel C_i)$ and then verifies whether h'_s is equal to h_s . If successful, the device computes $hd_i^* = h(D_i \parallel C_i) \oplus hd'_i$, the challenge $C_i^* = h(C_i \parallel k'_i)$ in

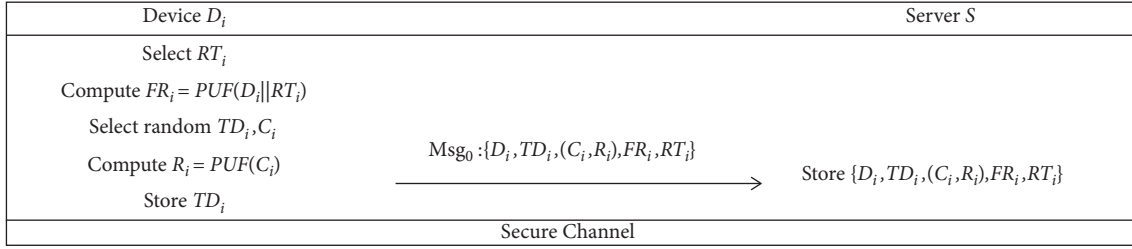


FIGURE 2: The registration phase.

the next round, the corresponding response $R_i^n = PUF(C_i^n)$, and $R_i^* = k_i' \oplus R_i^n$. Then, the device selects a nonce N_i , which is used to generate $N_i^* = k_i' \oplus N_i$, $h_i = h(C_i^n || R_i^n || k_i' || D_i || N_i)$ and the session key $sk = h(N_i || N_i^* || k_i')$. Next, the device stores $TD_i^n = h(TD_i || k_i')$ for the next round and sends $\text{Msg}_2: \{hd_i^*, R_i^*, N_i^*, h_i\}$ to the server.

- (3) After acquiring Msg_2 , the server computes the helper data $hd_i = h(D_i || C_i) \oplus hd_i^*$, the nonce $N_i' = k_i \oplus N_i^*$, the challenge $C_i^n = h(C_i || k_i)$, and its response $R_i^{n'} = k_i \oplus R_i^*$. Then, $h_i' = h(C_i^n || R_i^{n'} || k_i || D_i || N_i')$ is computed to verify the identity of h_i' and h_i . If the verification is passed, the server generates the session key $sk = h(N_i' || N_i || k_i)$ and the temporary pseudo-identity $TD_i^{n'} = h(TD_i || k_i)$ for the following round. Eventually, $\{TD_i^{n'}, (C_i^n, R_i^{n'})\}$ is kept in the database.

In summary, the procedure for an agreement of the session key between the physical device and the server in the IoT system is accomplished. The details are presented in Figure 3.

5.4. The Supplementary Subprotocol. If a desynchronization attack is launched when Msg_2 is sent to the server, the one-time temporary alias of the IoT device on the server end cannot be updated in time, which causes the messages of the IoT device and the server to be out of synchronization. In this regard, it is of vital necessity to introduce the supplementary subprotocol against the attack for the sake of the normal continuation of our authentication.

In the registration phase, the IoT device has calculated $FR_i = PUF(D_i || RT_i)$ and sent it to the server for storage. In the subprotocol phase shown in Figure 4, with the current timestamp T_i , the device computes $FR_i' = PUF(D_i || RT_i)$, $Fk_i^{n'} = h(D_i || RT_i || T_i) \oplus Fk_i^*$, and $Fk_i^* = h(D_i || RT_i || T_i) \oplus Fk_i^{n'}$ and then transmits $\text{Msg}_3 = \{Fk_i^*, Fhd_i^*, T_i, RT_i\}$ to the server end, which searches for the relevant data according to the registration time RT_i sent by the physical device and computes $Fk_i' = h(D_i || RT_i || T_i) \oplus Fk_i^*$, $Fhd_i' = h(D_i || RT_i || T_i) \oplus Fhd_i^*$ and $Fk_i = \text{Rec}(FR_i, Fhd_i')$ to compare Fk_i with $Fk_i^{n'}$ after receiving the message. If both are the same, the resynchronization is completed and the authentication process can continue normally.

6. Security Analysis

The BAN logic, designed by Burrows, Abadi, and Needham [30], features its simplicity and practicality, resulting in the general application to the formal security

analysis of identity verification protocols. However, even though it pioneered the formal analysis, its pitfalls were pointed out by Mao and Boyd [11]. Hence, we attempt to prove our proposed protocol to meet a series of requirements for the authentication between the IoT device and the server with the Mao and Boyd logic, namely, the improved BAN logic, in this section.

6.1. Basic Definitions. For the sake of eliminating negative features caused by the type mismatch, Mao and Boyd logic constructed three groups of type-specific objects, including principals, messages, and formulas, so we employ letters P and Q to describe principals, K , M , and N to represent messages, while X , Y , and Z symbolize formulas for the clarity and convenience [11].

Some definitions are listed below:

$$P | \equiv X, \quad (4)$$

$$P \stackrel{K}{|} \sim M, \quad (5)$$

$$P \stackrel{K}{|} \triangleleft M, \quad (6)$$

$$P \stackrel{K}{\longleftrightarrow} Q, \quad (7)$$

$$\#(N), \quad (8)$$

$$\text{sup}(P), \quad (9)$$

$$P \triangleleft || M. \quad (10)$$

Equation (4) denotes that principal P believes formula X to be true. Equation (5) shows that principal P says message M is encrypted with the key K . Equation (6) manifests that principal P sees message M is decrypted with key K . Equation (7) points out that K is considered as a good shared key between principals P and Q . Equation (8) suggests that message N is fresh that it has never appeared before the current protocol conducts. Equation (9) indicates that P is a super principal; namely, it is credible and legitimate. Equation (10) bespeaks that principal P cannot see the message M .

Considering the issue that the syntax is context-free while the relationship between messages is context-based, Mao and Boyd [11] explained that the idealization of

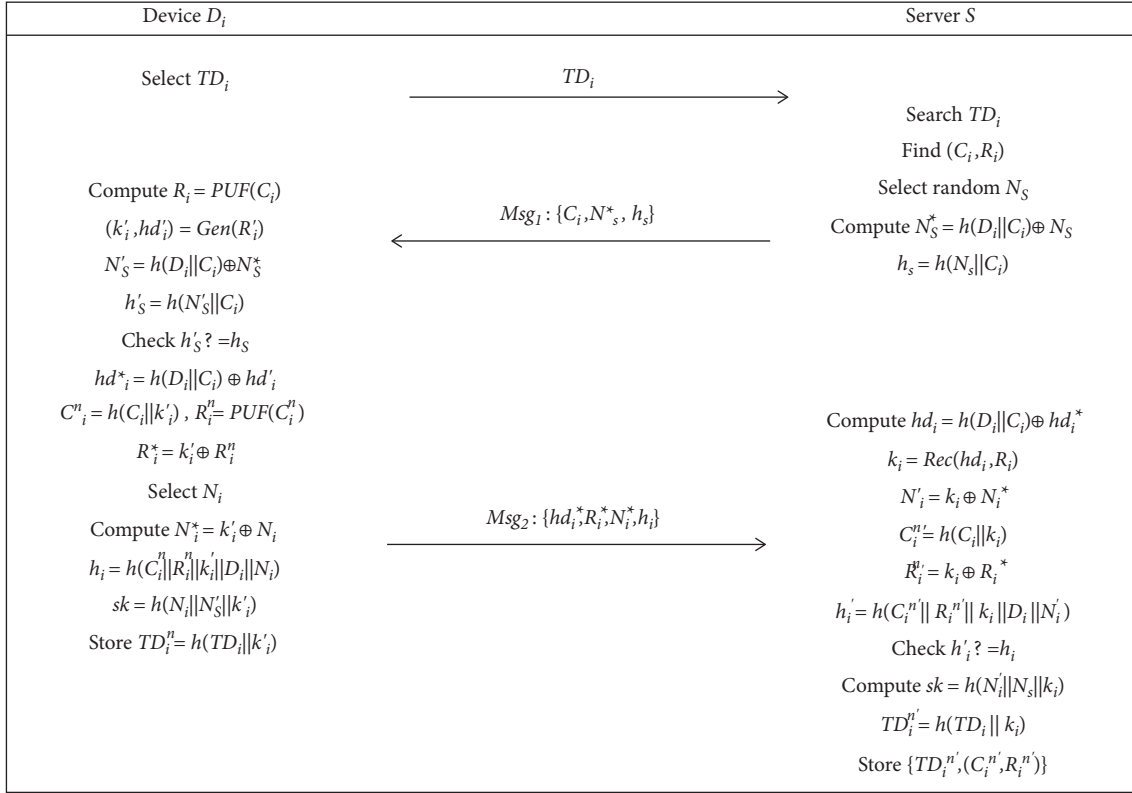


FIGURE 3: The authentication phase.

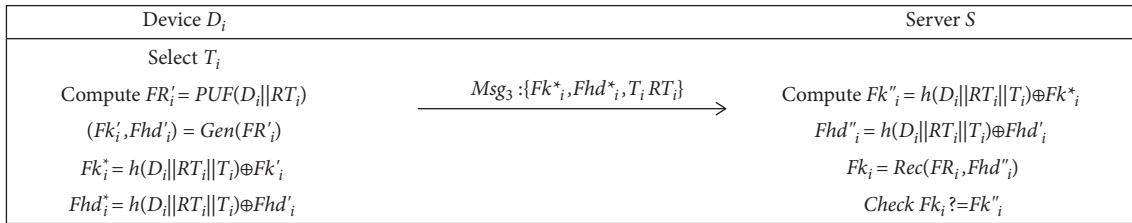


FIGURE 4: The supplementary subprotocol.

protocol messages converting the implicit contextual information to the explicit specification should be operated. There are some concepts of idealization regulations. On the one hand, there are five related concepts. The atomic message means a data unit with no symbols such as “,” “|”, “ \mathfrak{R} ”, “” or “”, in a message, where “,” is a combinator for a message and a principal, and “|” or “ \mathfrak{R} ” is a combinator for two messages. The challenge is an atomic message sent and received in two different lines by its originator, namely, a principal. In the meantime, the atomic message is not a timestamp. The replied challenge is a challenge existing in the message on the way to its originator. The response also belongs to the set of atomic messages excluding timestamps, which is sent with a replied challenge by its sender. If an atomic message is not a challenge, a response, or a timestamp, it is called nonsense. On the other hand, there are several idealization rules of messages in the protocol in the following:

- (i) All of the atomic messages considered as non-senses are supposed to be erased.
- (ii) If an atomic message plays both roles of the challenge and the response in a line, then it is regarded as a response.
- (iii) The challenges separated by commas can be combined with the symbol “|”, so do responses.
- (iv) The challenge and its corresponding response can be combined with the symbol “ \mathfrak{R} ”, whose form is “response \mathfrak{R} replied challenge”.
- (v) The message and its timestamp can also be combined with “ \mathfrak{R} ”, whose form is “message \mathfrak{R} timestamp”.

Moreover, according to [11], there are some inference rules which are created to achieve the intuitive formal analysis on the scheme of authentication and confidentiality in actual

applications, where symbol “ \wedge ” is a Boolean logic conjunction used to connect two formulas. For instance, if formula X and formula Y are true, then they can get the true formula Z , in the following form:

$$\frac{X \wedge Y}{Z}. \quad (11)$$

- (vi) The authentication rule (12): if P believes that K is a good shared key between P and Q and P sees M with K , P can believe Q encrypts M with K :

$$\frac{P| \equiv P \xleftrightarrow{K} Q \wedge P \triangleleft^K M}{P| \equiv Q| \sim M}. \quad (12)$$

- (vii) The confidentiality rule (13): there are three conditions: (1) P believes that K is a good key between P and Q ; (2) P believes that M cannot be obtained by anyone else; and (3) P can use K to encrypt the message M . If they are met, P can believe that only M can be available to P and Q :

$$\frac{P| \equiv P \xleftrightarrow{K} Q \wedge P| \equiv S^C \triangleleft \| M \wedge P| \sim M}{P| \equiv (S \cup \{Q\})^C \triangleleft \| M}. \quad (13)$$

- (viii) The nonce-verification rule (14): if P believes that M is fresh and that Q encrypts M with K , then P can believe that Q thinks K is a good key between P and Q :

$$\frac{P| \equiv \#(M) \wedge P| \equiv Q| \sim M}{P| \equiv Q| \equiv P \xleftrightarrow{K} Q}. \quad (14)$$

- (ix) The superprincipal rule (15): if P believes that Q trusts X and Q is a legitimate server, P can believe X :

$$\frac{P| \equiv Q| \equiv X \wedge P| \equiv \text{sup}(Q)}{P| \equiv X}. \quad (15)$$

- (x) The fresh rule (16): if P believes that M is fresh and P receives the message combined with N and M , P can believe that N is fresh:

$$\frac{P| \equiv \#(M) \wedge P \triangleleft N \mathfrak{R} M}{P| \equiv \#(N)}. \quad (16)$$

- (xi) The good-key rule (17): if P believes that K is not available to any other principal than P , and Q and K is fresh, P can believe that K is a good key between P and Q :

$$\frac{P| \equiv \{P, Q\}^C \triangleleft \| K \wedge P| \equiv \#(K)}{P| \equiv P \xleftrightarrow{K} Q}. \quad (17)$$

- (xii) The intuitive rule (18): it is a rule ignored usually that if P decrypts M with K , then P can see M :

$$\frac{P \triangleleft^K M}{P \triangleleft M}. \quad (18)$$

6.2. *Formal Security Analysis on Proposed Protocol.* According to the above inference rules, we propose some initial beliefs and assumptions for our protocol between the device and the server in the IoT system, which then are used to construct the security proofs.

Regarding the IoT device as D and the server as S , first, we try to prove the proposition (vi), which is “ S believes that N_s is a good shared key between S and D ”. As is shown in the following, (i) shows that S believes D_i is a good key between S and D because it is the real identity of the IoT device stored in the server; (ii) shows that S believes D_i cannot be known by any other one except D ; (iii) shows that S can encrypt N_s with D_i ; and (v) shows that S believes N_s is fresh because S generates the nonce N_s . In the light of the confidentiality rule, we use (i), (ii), and (iii) to obtain the statement “ S believes that no one else knows N_s except for S and D ”, which is (iv). Then, (iv) and (v) are applied in the good-key rule to get the final statement (vi). The detailed proof process is shown in Figure 5(a):

$$\begin{aligned} S| &\equiv S \xleftrightarrow{D_i} D (i), \\ S| &\equiv \#(N_s) (ii), \\ S| &\sim N_s (iii), \\ S| &\equiv \{S, D\}^C \triangleleft \| N_s (iv), \\ S| &\equiv \#(N_s) (v), \\ S| &\equiv S \xleftrightarrow{N_s} D (vi). \end{aligned} \quad (19)$$

Then, we attempt to prove the proposition (xvi), which is “ D believes that N_s is a good shared key between S and D ”. In the following, (vii) means D believes that D_i is a good shared key between D and S ; (viii) means that D can decrypt N_s with D_i ; (ix) means D believes that S encrypts N_s with D_i ; (x) means D believes that N_s is fresh; (xi) means D believes that S holds the belief that D_i is a good shared key between S and D ; (xii) means that D believes that S takes the belief that N_s cannot be known by others except for S ; (xiii) means D considers the fact that S believes only D and itself can obtain the nonce N_s ; and (xiv) means that D believes that S is a credible principal. Therefore, we can use these beliefs and assumptions to deduce the final conclusion. With the authentication rule, (vii) can be combined with (viii) to draw (ix). Additionally, (xi) can be derived from the combination between (ix) and (x) with the nonce-verification rule. With the three conditions (ix), (xi), and (xii) substituted into a variant of the confidentiality rule, we can reason out (xiii), which thereby together with (xiv) can be used in the superprincipal rule to obtain (xv). Then, (xv) and (x) are utilized to generate the final conclusion (xvi) with the good-key rule. The proof process is vividly shown in Figure 5(b):

$$\begin{array}{c}
\frac{S \models S \xleftrightarrow{D_i} D \wedge S \models D^c \triangleleft \| D_i \wedge S^{D_i} \| \sim N_S}{S \models \{S, D\}^c \triangleleft \| N_S} \wedge S \models \#(N_S)}{S \models S \xleftrightarrow{D} D} \\
\text{(a)}
\end{array}$$

$$\begin{array}{c}
\frac{D \models \#(N_S) \wedge \frac{D \models D \xleftrightarrow{D_i} S \wedge D \triangleleft N_S}{D \models S \models \#(N_S)} \wedge D \models S \models S^c \triangleleft \| N_S \wedge D \models S^{D_i} \| \sim N_S}{D \models S \models S \xleftrightarrow{D} D} \wedge D \models \text{sup}(S)}{D \models \{D, S\}^c \triangleleft \| N_S} \wedge D \models \#(N_S)}{D \models D \xleftrightarrow{D} S} \\
\text{(b)}
\end{array}$$

$$\begin{array}{c}
\frac{D \models D \xleftrightarrow{D_i} S \wedge D \models S^c \triangleleft \| k_i \wedge D^{k_i} \| \sim N_i}{S \models \{S, D\}^c \triangleleft \| N_i} \wedge D \models \#(N_i)}{D \models D \xleftrightarrow{D_i} S} \\
\text{(c)}
\end{array}$$

$$\begin{array}{c}
\frac{S \models \#(N_i) \wedge \frac{S \models S \xleftrightarrow{D_i} D \wedge S \triangleleft N_i}{S \models D \models \#(N_i)} \wedge S \models D \models D^c \triangleleft \| N_i \wedge S \models D \models \#(N_i)}{S \models D \models D \xleftrightarrow{D_i} S} \wedge S \models \text{sup}(D)}{S \models \{S, D\}^c \triangleleft \| N_i} \wedge S \models \#(N_i)}{S \models S \xleftrightarrow{D} D} \\
\text{(d)}
\end{array}$$

$$\begin{array}{c}
\frac{D \models D \xleftrightarrow{D_i} S \wedge D \models S^c \triangleleft \| k_i \wedge D \models R_i^n}{S \models \{S, D\}^c \triangleleft \| R_i^n} \wedge D \models \#(R_i^n)}{D \models D \xleftrightarrow{D_i} S} \\
\text{(e)}
\end{array}$$

$$\begin{array}{c}
\frac{S \models \#(N_i) \wedge \frac{S \models S \xleftrightarrow{D_i} D \wedge S \triangleleft N_i}{S \models D \models \#(N_i)} \wedge S \models D \models D^c \triangleleft \| R_i^n \wedge \frac{S \models S \xleftrightarrow{D_i} D \wedge S \triangleleft R_i^n}{S \models D \models \#(N_i)} \wedge S \models \text{sup}(D)}{S \models \{S, D\}^c \triangleleft \| R_i^n} \wedge S \models \#(N_i) \wedge \frac{S \triangleleft N_i \wedge R_i^n}{S \triangleleft N_i \wedge R_i^n}}{S \models S \xleftrightarrow{D} D} \\
\text{(f)}
\end{array}$$

FIGURE 5: (a) The proof for “S believes that N_S is a good shared key between S and D”. (b) The proof for “D believes that N_S is a good shared key between S and D”. (c) “D believes that N_i is a good shared key between D and S”. (d) “S believes that N_i is a good shared key between S and D”. (e) “D believes that R_i^n is a good shared key between D and S”. (f) “S believes that R_i^n is a good shared key between S and D”.

$$\begin{array}{l}
D \models D \xleftrightarrow{D_i} S \text{ (vii)}, \\
D \triangleleft N_S \text{ (viii)}, \\
D \models S \models \sim N_S \text{ (ix)}, \\
D \models \#(N_S) \text{ (x)}, \\
D \models S \models S \xleftrightarrow{D_i} D \text{ (xi)}, \\
D \models S \models S^c \triangleleft \| N_S \text{ (xii)}, \\
D \models S \models \{D, S\}^c \triangleleft \| N_S \text{ (xiii)}, \\
D \models \text{sup}(S) \text{ (xiv)}, \\
D \models \{D, S\}^c \triangleleft \| N_S \text{ (xv)}, \\
D \models D \xleftrightarrow{N_S} S \text{ (xvi)}.
\end{array} \tag{20}$$

Similarly, the proofs for “D believe that N_i is a good shared key between D and S” and “S believes that N_i is a good shared key between S and D” as, respectively, shown in Figures 5(c) and 5(d). In the matters of the former, according to the confidentiality rule, “D believes that k_i is a good shared key between itself and S”; “D believes that no one can obtain k_i except for S”; and “D encrypts N_i with k_i ”. These three conditions are involved in deducing a statement, which is “S holds the view that N_i can merely be known by S and D”. In the light of the conclusion, we can introduce it with the belief that “D believes N_i is fresh” into the good-key rule in order to obtain the final statement. Moreover, the latter is

generated by “S believing that N_i is fresh” which is the result of “S convinced that D believes only S and D can know N_i ”; “S believes that D is a legitimate principal” with the superprincipal rule; and “S believes that only S and D can obtain N_i ” with the good-share key rule. Obtained with the developed confidentiality rule, the statement “S is convinced that D believes only S and D can know N_i ” is the result of “S believing that D holds the belief that k_i is a good shared key between D and S”; “S is convinced that D believes that it is less likely for N_i to be attached by others except for D”; and “S believes that N_i is encrypted by D with k_i ”. In terms of the conclusion “S believes that D trusts k_i as a good shared key between D and S”. It can be deduced with the non-verification rule that “S believes N_i is a fresh nonce” and “S believes D can encrypt N_i with k_i ”, which can be obtained by the combination of “S believing that k_i is a good shared key between S and D” and “ N_i can be decrypted by S with k_i ” with the authentication rule.

In Figures 5(e) and 5(f), the similar manner of the proofs for “D believes that R_i^n is a good shared key between D and S” and “S believes that R_i^n is a good shared key between S and D” is described in the specific process. In Figure 5(e), with the confidentiality rule, we utilize three conditions: “D believes that k_i is a good shared key between D and S”; “D believes that no one can obtain k_i except for S”; and “ R_i^n can be encrypted by D with k_i ” to conclude the statement of “S believes it is impossible that a third person can obtain R_i^n except for S and D”, which is combined with the fact that “D believes R_i^n is fresh” to deduce the final belief of “D believes that R_i^n is a good shared key between D and S” with the good-

key rule. In Figure 5(f), what calls for special attention is that, with the fresh rule, the statement “S trusts R_i^n as fresh” is generated by “S believes that N_i is a fresh nonce” and “S can obtain N_i and R_i^n ”, which is concluded from “S can decrypt N_i and R_i^n with k_i^j ”, according to the intuitive rule.

In conclusion, generally, D_i is rarely known by others excluding D and S , so an adversary cannot obtain the secrets involved in the formal security proofs, which are N_S , N_i , R_i^n , and k_i^j . Some attacks like impersonation attacks are even less likely to be operated. Additionally, thanks to the feature of the PUF, they cannot get valid challenge-response pairs from it even when adversaries control an IoT device. Consequently, our protocol is regarded as reliable enough against some common security attacks.

7. Performance Analysis

In this section, we analyze the performance of the proposed scheme in three respects: security functions, computation costs, and communication costs, whose comparison results with the protocols in [10, 18, 21, 22] are introduced in the following.

7.1. Security Function Analysis. Aiming to present the strengths of the scheme proposed in the paper, we first compare it with four other PUF-based mutual authentication protocols on their security functions in Table 2, where $F_1, F_2, F_3, F_4, F_5, F_6, F_7, F_8$, and F_9 , respectively, represent the mutual authentication, the resilience to desynchronization, the impersonation attack, the session key security, the physical security, the reverse fuzzy extractor, the zero storage of shared secrets, the anonymity, and the lightweight feature. What is more, Y means achieved while N means not achieved.

In terms of resilience to desynchronization and the zero storage of the shared secrets, even when the scheme in [10] keeps a mass of alternate pseudonyms and keys, the desynchronization attack is still a problem. Although the protocol in [22] can prevent attacks to a certain degree, it still needs to store a large number of pseudo-identities and challenge-response pairs, which require a lot of storage space. According to the solution proposed in the paper, it is unnecessary for the IoT device and server to store those. When they are subjected to the desynchronization attack, they merely need to search for a subset in the database in the light of the registration time and finish the resynchronization. Moreover, the issue that it is more likely for noise to lead to some errors in the output is neglected by the scheme in [18]. While the scheme in [22] involves the fuzzy extractor, it does not reverse it to consider the resource imbalance between the device and server. Our scheme takes these factors into full consideration, and with the reverse fuzzy extractor, not only does it solve the noise problem, but it also takes reasonable advantage of resources. What is more, the protocol in [21] addresses the above issues, but it contains the public key cryptography, resulting in a surge of costs. Instead of it, our protocol is characterized by a series of lightweight functions, such as PUFs, hash functions, and

TABLE 2: The analysis of security functions.

Protocols	F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8	F_9
[18]	Y	Y	N	Y	Y	N	Y	N	Y
[10]	Y	N	Y	Y	Y	Y	N	Y	Y
[21]	Y	Y	Y	Y	Y	Y	Y	Y	N
[22]	Y	Y	Y	Y	Y	N	N	Y	Y
Our protocol	Y	Y	Y	Y	Y	Y	Y	Y	Y

XORs. Additionally, since the protocol in [18] directly uses the original identity of the device rather than its pseudo-identity, the anonymity is not achieved. Our resolve in the paper that uses the one-time temporary alias updated in each round of communication protects the privacy of the physical device in the IoT system.

7.2. Computation Costs Analysis. Considering the difference of the computation costs generated by various PUF-based protocols, we show the details in Table 3, where T_P, T_H, T_G, T_R , and T_S , respectively, symbolize the time costs of PUFs, hash functions (including the MAC), the key generation function of the fuzzy extractor, the reconstruction function of the fuzzy extractor, and symmetric encryption or decryption. Generally, we think that various time costs roughly meet the following magnitude relationships: $T_S > T_P \approx T_H$ and $T_R > T_G$.

Since the protocol in [21] is based on the three-party authentication, we just conduct the comparative analysis of our protocol and those in [10, 18, 22]. In our protocol, $h(D_i C_i)$ in the IoT device is used twice. As a result, we only consider the time cost of calculating it once. According to Table 4, we can conclude that our protocol still has a slight advantage compared with the protocol in [18]. Although it uses fewer hash functions, the time costs caused by the symmetric encryption and decryption with the response value bring our protocol the latest edge through a small victory. In addition, our protocol is one hash function less than that of [10], which is also a narrow margin. Furthermore, the computation costs of our PUFs and hash functions are similar to those of [22], but the device end equipped with the key generation function of the reverse fuzzy extractor costs fewer resources and less time.

7.3. Communication Costs Analysis. By analyzing the communication costs, we can still demonstrate some advantages of our proposed protocol. Since we regard l as a security parameter, utilizing the hash function to convert a bit string of arbitrary length into that of 1-bit length, we define the length of nonces, identities, challenge values, and response values as l bits, and the 1-bit data is changed to 8l-bit one after the symmetric encryption.

We contrast the computation costs of relevant protocols in [10, 18, 22], as shown in Table 4, attributing to the fact that the protocol in [21] involves three parties and causes numerous costs with asymmetric encryption and decryption. In Table 4, Size means the size of messages and Times means the times of sending messages. It is apparent that the computation costs of the protocol in [18] are much more

TABLE 3: The analysis of computation costs (ms).

Protocols	[18]	[10]	[22]	Our protocol
The IoT device end	$2T_P + 5T_H + 2T_S$	$2T_P + 7T_H + 1T_G$	$2T_P + 6T_H + 1T_R$	$2T_P + 6T_H + 1T_G$
The server end	$5T_H + 2T_S$	$7T_H + 1T_R$	$6T_H + 1T_G$	$6T_H + 1T_R$

TABLE 4: The analysis of communication costs (bits).

Protocols	[18]		[10]		[22]		Our protocol	
	Size	Times	Size	Times	Size	Times	Size	Times
The IoT device end	35 <i>l</i>	2	5 <i>l</i>	2	4 <i>l</i>	2	5 <i>l</i>	2
The server end	26 <i>l</i>	1	3 <i>l</i>	1	5 <i>l</i>	2	3 <i>l</i>	1
Total	61 <i>l</i>	3	8 <i>l</i>	3	9 <i>l</i>	4	8 <i>l</i>	3

TABLE 5: The summary comparisons of protocols.

Protocols	Security functions	Computation costs	Communication costs
[18]	Part	Highest	Highest
[10]	Part	Higher	Lowest
[21]	Part	—	—
[22]	Part	Higher	Higher
Our protocol	All	Lowest	Lowest

than any other protocol resulting from symmetric encryption and decryption. Additionally, the communication overhead of our protocol is as little as that in [10]. Besides, even though the communication costs of the IoT device in the protocol proposed by [22] are less than ours, regardless of the total size of messages or the total times of communications, the protocol in [22] is slightly more than ours. Therefore, our protocol in this paper can be treated low-overhead.

Above all, our protocol fully demonstrates its advantages in terms of security functions, computing costs, and communication overhead. Table 5 shows the summary comparisons among the protocols in [10, 18, 21, 22] and this paper. Since the computation and communication costs of the protocol in [21] are not involved in the above comparisons, we ignore them in Table 5, in which we can know that not only does our protocol meet all the security functions mentioned, but its computation and communication overhead is also the lowest.

8. Conclusion and Future Work

In this paper, we propose a lightweight and anonymous mutual authentication protocol for edge IoT nodes with physical unclonable functions. Instead of symmetric or asymmetric cryptography, the proposed protocol only uses lightweight operations, such as hash functions, PUFs, exclusive OR operations, and concatenation operations. On the one hand, we can solve the problem of a large number of pseudonyms in IoT devices due to anonymity and effectively resist physical security attacks from adversaries. On the other hand, we can consider PUF in nonideal environments and use fuzzy extractors to implement error correction to ensure the protocol's reliability. In addition, we present a

strict formal security proof to show that the proposed protocol meets the expected security requirements. Performance comparison analysis shows it has better computing efficiency and communication performance when compared with similar protocols.

We use subprotocols to resist desynchronization attacks. Although it is simple to implement, it is still not a very effective method to solve the desynchronization attack in the lightweight anonymous security authentication protocol. Therefore, our next work will further find better solutions.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they do not have any commercial or associative interest that represents a conflicts in connection with the work submitted.

Acknowledgments

The work was supported in part by the National Natural Science Foundation of China (61862052) and the Science and Technology Foundation of Qinghai Province (2020-ZJ-943Q).

References

- [1] M. El-hajj, A. Fadlallah, M. Chamoun, and A. Serhrouchni, "A survey of internet of things (iot) authentication schemes," *Sensors*, vol. 19, no. 5, 2019.

- [2] F. A. Turjman and M. Abujubbeh, "Iot-enabled smart grid via sm: an overview," *Future Generation Computer Systems*, vol. 96, pp. 579–590, 2019.
- [3] V. Chauhan, M. Patel, S. Tanwar, S. Tyagi, and N. Kumar, "Iot enabled real-time urban transport management system," *Computers & Electrical Engineering*, vol. 86, Article ID 106746, 2020.
- [4] B. S. Baker, W. Xiang, and I. Atkinson, "Internet of things for smart healthcare: technologies, challenges, and opportunities," *IEEE Access*, vol. 5, Article ID 26521, 2017.
- [5] S. H. Shah and I. Yaqoob, "A survey: internet of things (iot) technologies, applications and challenges," in *Proceedings of the 2016 IEEE Smart Energy Grid Engineering (SEGE)*, pp. 381–385, IEEE, Oshawa, ON, Canada, August 2016.
- [6] N. Bates, *Driverless Vehicle Security: Considering Potential Attacks and Countermeasures for Military Applications*, Department of Information Security, Egham, Surrey, 2020.
- [7] J. Cui, F. Wang, Q. Zhang, Y. Xu, and H. Zhong, "An anonymous message authentication scheme for semi-trusted edge-enabled iiot," *IEEE Transactions on Industrial Electronics*, vol. 68, Article ID 12921, 2020.
- [8] N. Tariq, A. Qamar, M. Asim, and F. A. Khan, "Blockchain and smart healthcare security: a survey," *Procedia Computer Science*, vol. 175, pp. 615–620, 2020.
- [9] J. Cui, J. Lu, H. Zhong, Q. Zhang, C. Gu, and L. Liu, "Parallel key-insulated multi-user searchable encryption for industrial internet of things," *IEEE Transactions on Industrial Informatics*, 2021.
- [10] P. Gope and B. Sikdar, "Lightweight and privacy-preserving two-factor authentication scheme for iot devices," *IEEE Internet of Things Journal*, vol. 6, no. 1, pp. 580–589, 2018.
- [11] W. Mao and C. Boyd, "Towards formal analysis of security protocols," in *Proceedings of the Computer Security Foundations Workshop VI*, pp. 147–158, IEEE, Franconia, NH, USA, June 1993.
- [12] I. Lee and K. Lee, "The internet of things (iot): applications, investments, and challenges for enterprises," *Business Horizons*, vol. 58, no. 4, pp. 431–440, 2015.
- [13] M. M. Fouda, Z. M. Fadlullah, N. Kato, R. L. Rongxing, and X. S. S. Xuemin, "A lightweight message authentication scheme for smart grid communications," *IEEE Transactions on Smart Grid*, vol. 2, no. 4, pp. 675–685, 2011.
- [14] P. Porambage, C. Schmitt, P. Kumar, A. Gurtov, and M. Ylianttila, "Two-phase authentication protocol for wireless sensor networks in distributed iot applications," in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC)*, pp. 2728–2733, Istanbul, Turkey, April 2014.
- [15] R. Amin, S. K. Islam, M. K. Khan, A. Karati, D. Giri, and S. Kumari, "A Two-Factor Rsa-Based Robust Authentication System for Multiserver Environments," *Security and Communication Networks*, vol. 2017, Article ID 5989151, 15 pages, 2017.
- [16] A. K. Das, P. Sharma, S. Chatterjee, J. K. Sing, and K. S. Jamuna, "A dynamic password-based user authentication scheme for hierarchical wireless sensor networks," *Journal of Network and Computer Applications*, vol. 35, no. 5, pp. 1646–1656, 2012.
- [17] M. Turkanovic and M. Holbl, "An improved dynamic password-based user authentication scheme for hierarchical wireless sensor networks," *Elektronika ir Elektrotehnika*, vol. 19, no. 6, pp. 109–116, 2013.
- [18] M. N. Aman, K. C. Chua, and B. Sikdar, "Mutual authentication in iot systems using physical unclonable functions," *IEEE Internet of Things Journal*, vol. 4, no. 5, pp. 1327–1340, 2017.
- [19] U. Chatterjee, V. Govindan, R. Sadhukhan et al., "Building puf based authentication and key exchange protocol for iot without explicit crps in verifier database," *IEEE Transactions on Dependable and Secure Computing*, vol. 16, pp. 424–437, 2018.
- [20] K. B. Frikken, M. Blanton, and M. J. Atallah, "Robust authentication using physically unclonable functions," in *Proceedings of the International Conference on Information Security*, pp. 262–277, Springer, Pisa, Italy, September 2009.
- [21] Qi Jiang, X. Zhang, N. Zhang, Y. Tian, X. Ma, and J. Ma, "Two-factor authentication protocol using physical unclonable function for iov," in *Proceedings of the 2019 IEEE/CIC International Conference on Communications in China (ICCC)*, pp. 195–200, Changchun, China, October 2019.
- [22] P. Gope, J. Lee, and T. Q. S. Quek, "Lightweight and practical anonymous authentication protocol for rfid systems using physically unclonable functions," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 11, pp. 2831–2843, 2018.
- [23] W. Feng, Y. Qin, S. Zhao, and D. Feng, "Aaot: lightweight attestation and authentication of low-resource things in iot and cps," *Computer Networks*, vol. 134, pp. 167–182, 2018.
- [24] M. Mitev, M. H. Shekiba, A. Chorti, and M. Reed, "Multi-factor physical layer security authentication in short block-length communication," 2020, <https://arxiv.org/abs/2010.14457>.
- [25] R. Maes, *Physically Unclonable Functions: Constructions, Properties and Applications*, Katholieke Universiteit Leuven, Leuven, Belgium, 2012.
- [26] G. E. Suh and S. Devadas, "Physical unclonable functions for device authentication and secret key generation," in *Proceedings of the 2007 44th ACM/IEEE Design Automation Conference*, pp. 9–14, San Diego, CA, USA, June 2007.
- [27] Y. Dodis, L. Reyzin, and A. Smith, "Fuzzy extractors: how to generate strong keys from biometrics and other noisy data," in *Proceedings of the International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 523–540, Springer, Interlaken, Switzerland, May 2004.
- [28] A. V. Herrewewege, S. Katzenbeisser, R. Maes et al., "Reverse fuzzy extractors: enabling lightweight mutual authentication for puf-enabled rfids," in *Proceedings of the International Conference on Financial Cryptography and Data Security*, pp. 374–389, Springer, Kralendijk, Bonaire, Sint Eustatius and Saba, March 2012.
- [29] D. Dolev and A. Yao, "On the security of public key protocols," *IEEE Transactions on Information Theory*, vol. 29, no. 2, pp. 198–208, 1983.
- [30] M. Burrows, M. Abadi, and R. N. Michael, "A logic of authentication," *Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences*, vol. 426, pp. 233–271, 1989.

Research Article

A Virtual Machine Migration Strategy Based on the Relevance of Services against Side-Channel Attacks

Ji-Ming Chen ^{1,2} Shi Chen ¹ Xiang Wang ¹ Lin Lin ¹ and Li Wang ¹

¹School of Computer Science and Communication Engineering, Jiangsu University, Zhenjiang, China

²Key Laboratory of Security Technology for Industrial Cyberspace, Jiangsu University, Zhenjiang, China

Correspondence should be addressed to Ji-Ming Chen; jmchen@ujs.edu.cn

Received 31 August 2021; Accepted 29 November 2021; Published 21 December 2021

Academic Editor: Jie Cui

Copyright © 2021 Ji-Ming Chen et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

With the rapid development of Internet of Things technology, a large amount of user information needs to be uploaded to the cloud server for computing and storage. Side-channel attacks steal the private information of other virtual machines by coresident virtual machines to bring huge security threats to edge computing. Virtual machine migration technology is currently the main way to defend against side-channel attacks. VM migration can effectively prevent attackers from realizing coresident virtual machines, thereby ensuring data security and privacy protection of edge computing based on the Internet of Things. This paper considers the relevance between application services and proposes a VM migration strategy based on service correlation. This strategy defines service relevance factors to quantify the degree of service relevance, build VM migration groups through service relevance factors, and effectively reduce communication overhead between servers during migration, design and implement the VM memory migration based on the post-copy method, effectively reduce the occurrence of page fault interruption, and improve the efficiency of VM migration.

1. Introduction

With the development of the Internet of Things technology, the number of Internet of Things devices is increasing rapidly, which means that a large amount of data will be generated for processing and storage. Due to the limited computing and storage capabilities of IoT devices, these data are usually uploaded to a cloud server for processing. However, the long-distance data transmission of ordinary cloud computing is difficult to meet the needs of resource-intensive and delay-sensitive IoT applications [1]. As a result, edge computing was created. Edge computing delivers ultra-low latency and high bandwidth for IoT devices to satisfy the data processing and storage requirements by putting computing and storage resources on the network edge near the IoT devices [2, 3]. The massive data generated by the Internet of Things devices in the edge computing-based IoT will contain a large amount of user identity information, location information, and sensitive information, so leakage of user information will inevitably pose significant security risks to users [4].

In edge computing, the most common method of information leakage is through side-channel attack. Malicious users use the underlying shared resources of the server to build a side channel, bypassing the logical isolation provided by the virtualized environment, and stealing the private information of other coresident virtual machines [5]. Rather than focusing on the mapping between plaintext and ciphertext, side-channel attacks obtain keys by analyzing nonfunctional behaviors and encryption or decryption operations [6, 7]. As a result, the commonly used strong encryption schemes cannot avoid exposing this physical information, posing a significant threat to the data security and privacy protection of edge computing-based IoT. Therefore, defense against side-channel attacks is crucial for edge computing. At the moment, virtual machine migration technology is frequently used to defend against side-channel attacks, which is the primary method of defense [8, 9]. Compared with traditional countermeasures, the VM migration method is general and immediately deployable [10]. By dynamically migrating virtual machines, the time for

coexistence between virtual machines is reduced, thereby reducing the amount of information that an attacker steals from the target virtual machine, so that the attacker cannot successfully obtain the target's information. Virtual machine migration can effectively prevent attackers from realizing coresident virtual machines, thereby ensuring data security and privacy protection of edge computing-based IoT [11].

As the types of IoT services increase, and their functions become more complex, a service request often requires a combination of services on multiple virtual machines. Therefore, if the association between services is not considered in the process of virtual machine migration, it will bring a large amount of communication overhead between servers and increase network energy consumption to the cloud data center. Therefore, this paper designs a virtual machine migration strategy based on service relevance, defines server relevance factors to quantify the degree of relevance between services, and migrates more relevant virtual machines to the same target server according to the degree of relevance between services. A post-copy method based on service priority is also designed to minimize the total migration time, and page fault rate. The experiment proved that the virtual machine migration strategy based on service relevance minimizes the communication overhead between servers while achieving fast and efficient VM migration.

2. Related Work

Virtual machine migration technology mainly involves server resource monitoring, load forecasting, virtual machine placement methods, and memory migration execution. In the process of VM live migration, firstly select the VM to migrate and the target server through the virtual machine placement method in the migration scheduling phase, and then complete the VM migration through the memory migration method in the migration execution phase. At present, there are many research studies on VM migration technology, which mainly focus on VM placement method and VM memory migration method.

2.1. VM Placement Methods. Virtual machine placement methods are mainly divided into migration time selection, virtual machine selection, and target server selection.

For the migration time selection problem, the prediction method is generally used to predict the resource utilization rate of the server in the future. For example, Melhem et al. [12] proposed a host load detection algorithm to determine overload or light load and migrate VMs to achieve server consolidation or load balancing.

For the VM selection and target server selection problem, Sotiriadis et al. [13] proposed an adaptive VM scheduling algorithm. The algorithm selects the VM to migrate by analyzing real-time resource monitoring data and migrates the VM to the server with the highest server evaluation value. In order to reduce the communication overhead caused by migration, Liu et al. [14] proposed a correlation-based VM migration algorithm to quantify the

relationship between the resource requirements of VMs and time-varying resources, build the VM group with the highest relevance as the migration unit, and select the server with the least relevance to the virtual group as the target server. Rajabzadeh and Haghghat [15] selected the VM with the highest CPU utilization without sacrificing SLA and used the Markov chain model to select the target server. In order to reduce the resource consumption caused by migration, Xu and Fortes [16] considered the total resource waste, power consumption, and cooling cost of the VM when it was running. However, the relationship between VMs and the cost of live migration is not fully considered. In literature [17], Verma et al. considered energy consumption and migration costs, but their research showed that it is difficult to estimate the exact power consumption of the server. A novel VM placement algorithm [18] designed a new technology called resource usage factor, which can be used to quantify server resource usage and place VMs on suitable physical machines to improve the resource utilization of physical machines. In addition, Kanniga Devi et al. [19, 20] also did research on optimizing placement methods. They used cluster intelligent algorithms to optimize the selection of VMs or target servers in VM migration strategies, reducing energy consumption in cloud computing and improving Resource utilization rate.

2.2. VM Memory Migration Methods. The research of memory migration methods mainly focuses on the pre-copy method and the post-copy method [21].

Mandal et al. [22] designed an algorithm to find the appropriate bandwidth and the number of prereplication iterations. It develops a model to measure network resource consumption, migration time, and downtime and determine the appropriate migration bandwidth and number of prereplication iterations to improve performance. In order to reduce the total migration data, the paper [23] proposed a method to optimize the pre-copy method, which can reduce unnecessary memory page transfer, and the feature-based compression (CBC) algorithm reduces the total migration time and downtime. In order to ensure that the memory migration is completed within a specific time, Zhang et al. [24] proposed a novel transmission control mechanism to ensure the bandwidth of the calculation and theoretically analyzed the bandwidth demand to ensure the total migration time and downtime. Literature [25] conducts a comprehensive empirical study on the pre-copy method to provide suggestions for optimal selection. To optimize post-copy method, Su et al. [26] improved the subsequent copy method by eliminating unnecessary remote page errors.

Sun et al. [27] proposed an improved serial migration strategy, which is based on the post-copy method and supports multivirtual machine migration. Lei et al. [28] proposed a hybrid copy method, which combines the pre-copy method with the subsequent copy method to make up for the defects of the pre-copy method and the post-copy method, but this method still has page fault. Deshpande et al. [29] designed an eviction-aggregation VM live migration method. It speeds up the eviction time of the VM migration

process by setting the cache area and restores the overloaded server to the normal state in the fastest time.

At present, when selecting multiple VMs for migration, the research on VM selection seldom considers the communication consumption between the VMs. If closely related VMs are migrated to different servers, it will bring more data transmission overhead between servers in the calculation process. This paper defines service relevance factor to quantify the degree of relevance between services and proposes a VM selection strategy based on service relevance. It combines closely connected VMs into a VM migration group for migration to reduce energy consumption and communication overhead. For memory migration methods, this paper chooses to improve the post-copy method and proposes a post-copy memory migration method based on service priority (PBSCP). The method sets the initial priority of the service based on service relevance and updates the service priority according to the page fault situation to reduce the occurrence of page fault interruption. This method also reduces the migration time and the migration data volume by adding temporary storage devices and reduces the occurrence of page faults.

3. The Design of Virtual Machine Migration Strategy

3.1. Service Relevance Factor. Figure 1 describes the communication relationship between devices in the Edge Computing. Users and IoT devices upload the collected data to the edge server through the network for processing and storage, and then the VMs on the server provide services to users and IoT devices, so the edge server will generate interserver communication overhead due to the cooperation between services. The dynamic nature of multiple service relationships affects the communication overhead between servers in the process of VM migration. Before designing VM selection and memory migration strategies based on service relevance, we need to quantify the degree of association between services, which determines the value of the service relevance factor.

This paper defines the service relevance factor as the degree of relevance between application service programs on each VM. Before measuring the degree of association, the interaction rate between application services and the communication overhead of each interaction need to be considered. The service set is defined as $I = \{I_1, I_2 \dots I_k \dots I_n\}$, and there are different interaction rates and communication overheads between services.

Assume that service I_i on virtual machine V_k and service I_j on virtual machine V_l have an interaction relationship when providing services to users. The interaction rate factor is defined as $IR_f^{ij}(V_{kl})$, which represents the number of

interactions required to process service requests per second. According to the interaction factor, define the server overhead $SC(I_i, I_j)$ consumed by two related services as

$$SC(I_i, I_j) = IR_f^{ij}(V_{kl}) \cdot \Delta c_{ij}, \quad (1)$$

where Δc_{ij} represents the cost of processing each interactive task.

Since a VM can contain multiple services, the relationship between services is also different. If there is no interaction between services, the service overhead is 0. Therefore, the calculation formula of service relevance factor can be expressed as

$$SC(I_i, I_j) = \begin{cases} IR_f^{ij}(V_{kl}) \cdot \Delta c_{ij}, & I_i \text{ is related to } I_j, \\ 0, & \text{others.} \end{cases} \quad (2)$$

3.2. VM Selection Strategy Based on Service Relevance. A virtual machine migration strategy is based on the relevance of services against side-channel attacks, a VM group with a higher association is built through the service relevance factor, and the VM migration group is used as a whole for migration. From the perspective of service relevance, the VM migration group is a closely connected “area” on the server. Migrating this whole to the target server can improve task execution efficiency and reduce network communication overhead between servers.

During the construction and expansion of the VM migration group, the VM with the highest relevance within the group should be found. Therefore, in the process of building a VM migration group, it is necessary to compare the service relevance C^{GI} between a certain VM and the migration group, and the service relevance C^{GO} between this VM and outside the group.

This paper defines $CF(V_i, G)$ to indicate the association between VM V_i and VM migration group G . The calculation of $CF(V_i, G)$ can be expressed as

$$CF(V_i, G) = C_{V_i, G}^{GI} - C_{V_i, G}^{GO}. \quad (3)$$

According to the calculation formula of service relevance factor (2), it can be seen that the server cost between any two VMs is shown in the following formula:

$$\begin{aligned} C^L(V_i, V_k) &= \sum_{I_x \in V_i} \sum_{I_y \in V_k} SC(I_x, I_y) \\ &= \sum_{I_x \in V_i} \sum_{I_y \in V_k} IR_f^{xy}(V_{ik}) \cdot \Delta c_{xy}. \end{aligned} \quad (4)$$

Formula (4) can be further expressed as

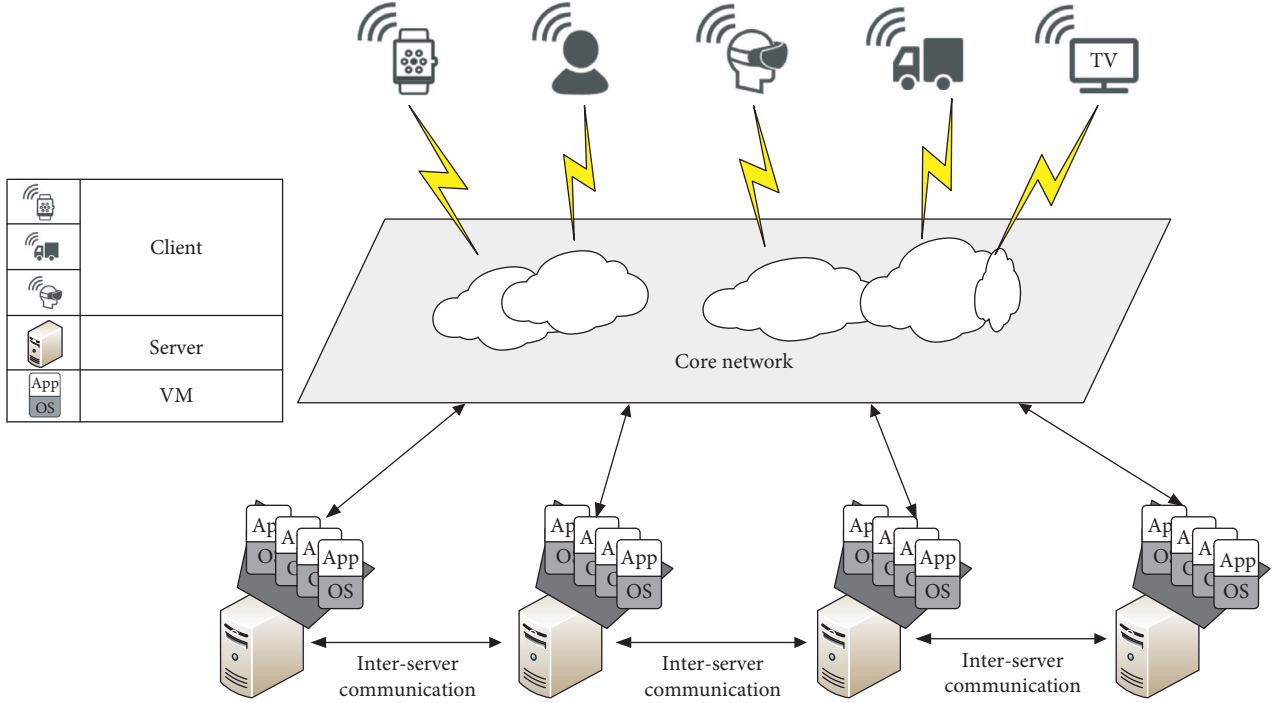


FIGURE 1: Communication relationship between devices in the edge computing-based IoT.

$$\begin{aligned}
 CF(V_i, G) &= \sum_{V_k \in G} (V_i, V_k) - \sum_{V_l \notin G} (V_i, V_l) \\
 &= \sum_{V_k \in G} \sum_{I_x \in V_k} \sum_{I_y \in V_i} IR_f^{xy}(V_{ki}) \cdot \Delta c_{xy} - \sum_{V_l \notin G} \sum_{I_z \in V_l} \sum_{I_y \in V_i} IR_f^{zy}(V_{li}) \cdot \Delta c_{zy}.
 \end{aligned} \tag{5}$$

Set the server overload threshold $S_{\text{threshold}}$. If the service load exceeds the threshold, it will be regarded as an overload state, in order to facilitate the measurement of the load condition that the server can accept when it is not overloaded. The load capacity of the server S_i is defined as W_i^{accept} .

In this paper, the VM selection strategy is the construction process of the VM migration group. When determining the amount of migration, find the VM with the largest load on the original server as the initial migration group, and then expand the initial migration group to include more load.

During the expansion process, the VM with the closest relationship to the group must be selected each time, and the load of this VM and the load of the migration group must be calculated whether the total load reaches the migration data volume. If the required volume of migration data is not reached, the VM is added to the group; otherwise, the next closest VM is selected, and the process is repeated until the load of the migration group meets the requirements. The specific strategy is shown in Figure 2.

Assuming that the load to be migrated for original server S_i is W , the acceptable load difference threshold is W_{bound} ,

the VM on S_i is denoted as $\{V_1, V_2 \dots V_m\}$, the VM migration group constructed is $G = \{V_1, V_2 \dots V_k\}$, and the construction algorithm of the VM migration group is as shown in Algorithm 1.

3.3. Memory Migration Strategy Based on Service Priority. The VM migration group constructed based on service relevance has the characteristics of large data volume and many service requests. This paper combines service relevance and designs a post-copy memory migration method (PCBSP) based on service priority. The method uses service priority to determine the corresponding memory page push priority and adds temporary storage devices (TD) during the migration process to make the migration time minimize and reduce the page fault rate.

3.3.1. Overall Process of PCBSP. In order to minimize the VM migration time and reduce the page fault rate, PCBSP adds a memory page push algorithm based on service priority and temporary storage devices on the basis of the post-copy method. The basic process of PCBSP is shown in Figure 3.

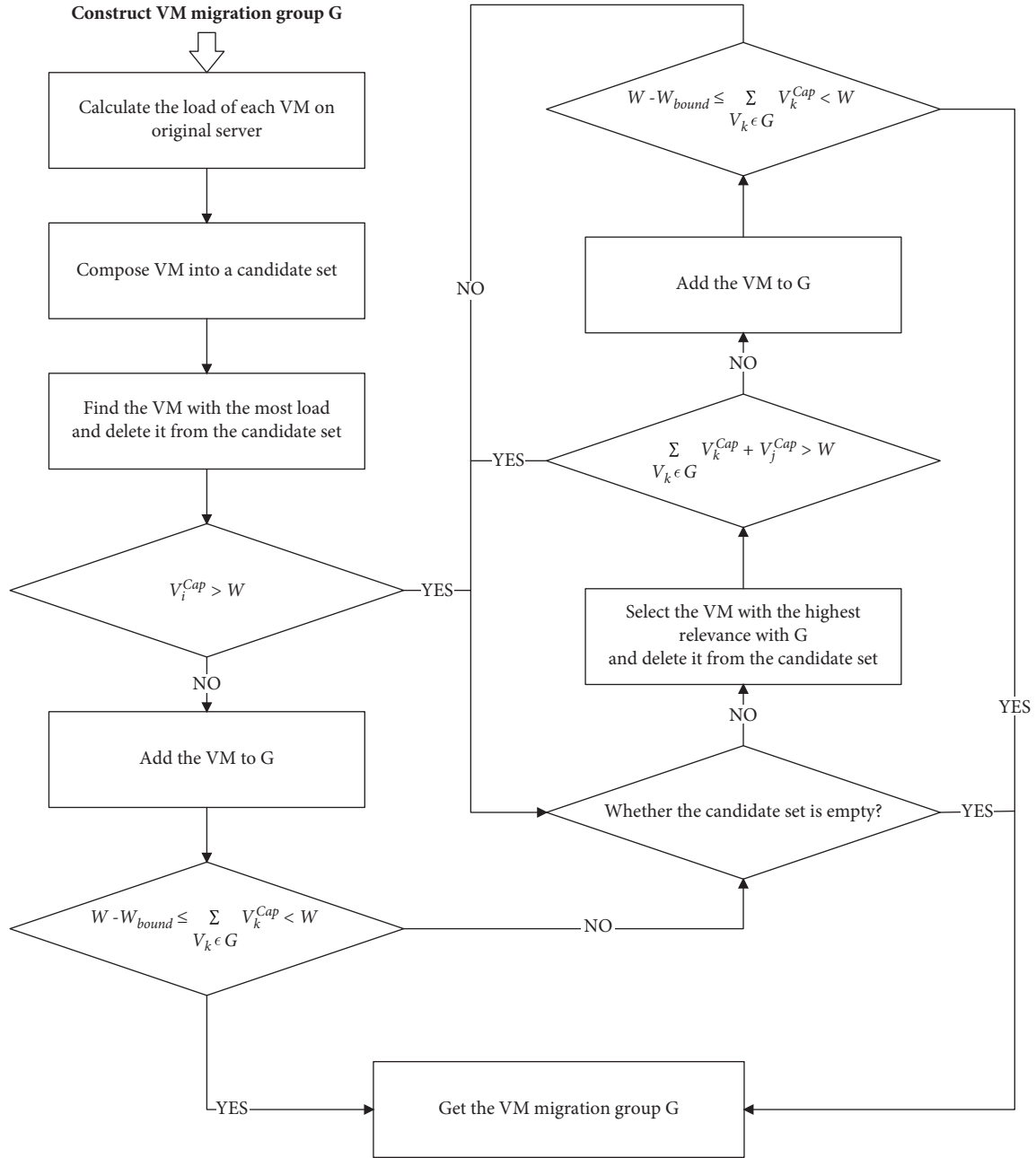


FIGURE 2: Process of building a VM migration group.

PCBSP mainly includes three stages. The first phase needs to select the appropriate spare memory construction in the physical server. The second phase is the VM migration phase, which includes migrating memory pages to $T D$ and sending memory pages directly to the target server.

The third phase is the VM migration phase, which includes the migration of memory pages from $T D$ and the direct acquisition of required memory pages from the original server. The migration of memory pages is collectively called the memory page push.

3.3.2. *Priority Calculation.* Since multiple services can be deployed on a VM, that is, the VM migration group contains several services, and each service provides a service when the

user requests it, there may be dependent or dependent relationships among multiple services. In this paper, a memory page push algorithm (PBSP) based on service priority is designed to actively push memory pages to reduce the occurrence of page fault in the target server.

When the VM is resumed on the target server, the initial priority of the service and related pages is calculated according to the service request rate and the dependency relationship between the services.

Definition 1 (service rate). For a certain service I_j in the VM migration group G , the number of requests received from users per unit time is taken as the service rate, which is recorded as $SR(I_j)$.

Input: server load information
Output: VM migration group G
Method:

- (1) Analyze the server load information to get the original server as S_i ;
- (2) Get the set of virtual machines on S_i as AllVmList, set to $\{V_c^*\}$;
- (3) Calculate the VM load on S_i , mark it as $\{V_i^{Cap}\}$, and sort in descending order of load;
- (4) Initialize the VM migration group G;
- (5) **while**($\{V_c^*\} \neq \text{null}$) **do**
- (6) find the VM with $\max(V_i^{Cap})$ and set it as V_j ;
- (7) $\{V_c^*\} = \{V_c^*\} / V_j$;
- (8) **if** ($V_i^{Cap} < W$) **then**
- (9) add V_j to G;
- (10) **if** ($W - W_{\text{bound}} \leq V_i^{Cap} < W$) **then**
- (11) **return** VM migration group G;
- (12) **else**
- (13) **break**;
- (14) **end if**
- (15) **end if**
- (16) **end while**
- (17) **while**($\{V_c^*\} \neq \text{null}$) **then**
- (18) select $\{V_c^*\}$ in $\max(CF(V_j, G))$, that is, select the VM with the most service relevance and set it to V_j ;
- (19) $\{V_c^*\} = \{V_c^*\} / V_j$;
- (20) **if**($\sum_{V_k \in G} V_k^{Cap} + V_j^{Cap} < W$) **then**
- (21) add V_j to G;
- (22) **if** ($W - W_{\text{bound}} \leq \sum_{V_k \in G} V_k^{Cap} + V_j^{Cap} < W$) **then**
- (23) **break**;
- (24) **else**
- (25) **continue**;
- (26) **end if**
- (27) **end if**
- (28) **end while**
- (29) **return** VM migration group G;

ALGORITHM 1: Build VM migration group.

Figure 4 is an example graph of an associated service group. Services are represented by dots in the graph, and service dependencies are represented by directed edges in the graph.

Definition 2. According to the dependency between services in the associated service group, define the set of service nodes, which is a dependence of service I_j as $\text{reply}(I_j)$ and the set of service nodes, which is dependent on I_j as $\text{depended}(I_j)$.

Set the default priority of Service I_j as

$$DR(I_j) = \alpha \frac{SR(I_j)}{\sum_{I_i \in G} SR(I_i)}, \quad (6)$$

where α represents a parameter that can simplify $DR(I_j)$ to an integer.

Each service performs related functions and needs to load the corresponding program into the memory.

Through this loading mechanism, the corresponding page fault rate can be inferred according to the probability of service access. Therefore, when calculating the preset priority, the service priority and the related memory page priority can be equivalent. Thus, the default priority of memory pages related to I_j can be obtained:

$$DR(\text{pages}(I_j)) = DR(I_j). \quad (7)$$

For a service, which is being dependent on, the greater its out-degree, the greater the $|\text{depended}(I_j)|$, and the priority push of its related pages can prevent page faults due to the lack of dependent services when users request the service, which is being dependent on, which can effectively reduce the page fault rate. This paper comprehensively considers the in-degree and out-degree of the service and defines the initial priority calculation method for the service and its related pages:

$$\begin{aligned} PR(\text{pages}(I_j)) &= PR(I_j) \\ &= DR(I_j) \cdot (\beta_1 |\text{depended}(I_j)| + \beta_2 |\text{reply}(I_j)|) \\ &= \alpha \frac{SR(I_j)}{\sum_{I_i \in G} SR(I_i)} \cdot (\beta_1 |\text{depended}(I_j)| + \beta_2 |\text{reply}(I_j)|). \end{aligned} \quad (8)$$

For page fault service I_k , when the VM resumes running on the target server, first check whether there are any memory pages for service I_k , and if so, push the memory pages according to the initial priority calculated by (8).

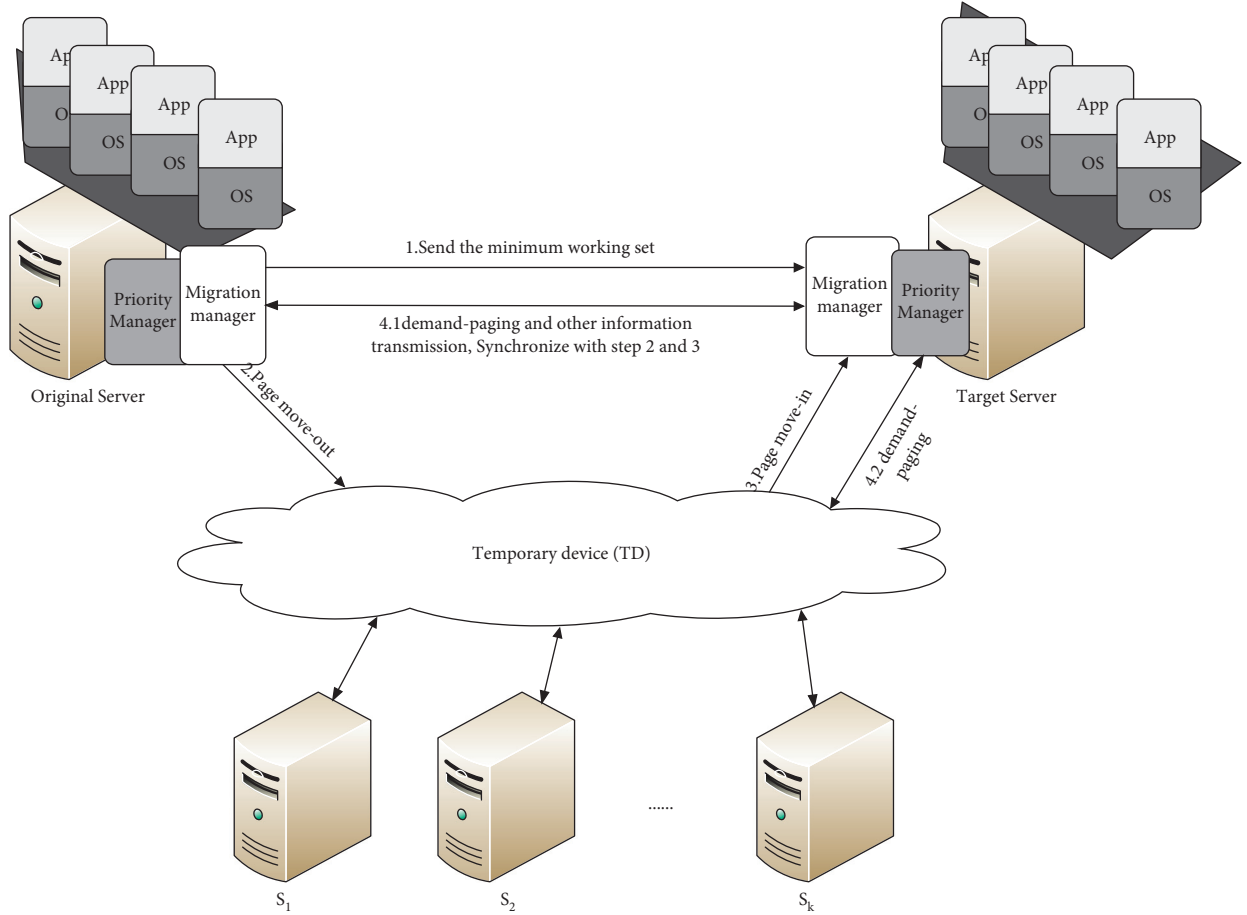


FIGURE 3: Overall process of PCBSP.

The priority of memory page push is mainly determined by $|\text{reply}(I_j)|$, $|\text{depended}(I_j)|$, and $SR(I_j)$. Since $|\text{reply}(I_j)|$ and $|\text{depended}(I_j)|$ are fixed amounts, the service rate $SR(I_j)$ is an average value for a period of time, which is not related to the interaction sequence. If the interaction rate is low, but the access is earlier, according to the priority, it may not be pushed first, and page faults will eventually occur. Therefore, the priority needs to be dynamically adjusted according to the real-time situation of service access during the memory migration process.

In order to dynamically adjust the priority, this paper introduces the following parameters: V_0 , T , push_tab1 and push_tab2 , where V_0 represents the change value of the priority related to page faults when the original server receives a page fault request, T represents the priority update cycle, and push_tab1 represents the memory page priority array that has not yet been migrated to $T D$, and push_tab2 represents the memory page priority array on $T D$.

The push_tab1 and push_tab2 priority update methods are the same, and the two arrays are collectively referred to as push_tab .

Set the index of memory page in the priority array push_tab as push_index , service I_j of page, and if there is no

page fault request related to service I_j within T , the priority is reduced as

$$\begin{aligned} & \text{push_tab}[\text{page_index}] \\ &= \frac{\text{push_tab}[\text{page_index}] + \text{PR}(\text{pages}(I_j))}{2}. \end{aligned} \quad (9)$$

If a page fault request for a service related to page occurs within T , the page faulted service is I_k . If this service is a dependence of the service in page, the priority is increased to

$$\begin{aligned} & \text{push_tab}[\text{page_index}] \\ &= \text{push_tab}[\text{page_index}] + \frac{V_0}{|\text{depended}(I_k)|}. \end{aligned} \quad (10)$$

If this service is dependent on the service in page, the priority is increased to

$$\begin{aligned} & \text{push_tab}[\text{page_index}] \\ &= \text{push_tab}[\text{page_index}] + V_0. \end{aligned} \quad (11)$$

Because once a page fault occurs, as a dependence service, there is a high probability of a page fault.

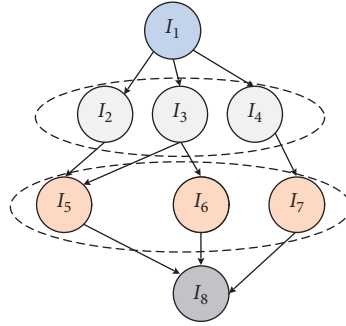


FIGURE 4: Example of associated service group.

3.3.3. Implementation of PBSP. When a large number of page faults occur in the restored VM, it directly requests memory pages from the original server. At this time, the page fault transmission is directly transmitted from the original server to the target server in descending order of the initial priority. The remaining memory pages in `push_tab` adjust their priority dynamically and actively push memory pages to `TD` in descending order of priority.

The memory page push algorithm (PBSP) based on service priority is given in Algorithm 2.

4. Implementation

This section mainly introduces the overall process of the VM migration strategy based on service relevance and introduces the VM selection strategy based on service relevance in the migration scheduling phase and the specific implementation of the memory migration method based on service priority in the memory migration phase.

4.1. Overall Process. The overall process of VM live migration is divided into two phases, namely, the migration scheduling phase and the migration execution phase. In this paper, the VM selection strategy based on service relevance is applied to the migration scheduling module, and the memory migration method based on service priority is applied to the migration execution module. The overall framework of VM live migration is shown in Figure 5.

The main process is divided into two phases: the migration scheduling phase and the migration execution phase. The migration scheduling phase mainly involves VM selection and the target server selection, and the migration execution phase mainly involves memory migration.

4.2. Implementation of Migration Scheduling Phase. The migration scheduling phase is mainly divided into two parts. Firstly, the load that needs to be migrated is calculated according to the original server's own load information and builds the VM migration group. Then, the target server is selected to place the VM migration group according to the load information of other servers provided by the VM migration management center. The selection of VMs is the key to migration scheduling. This paper uses a VM selection strategy based on service relevance to construct a VM

migration group. The process of the migration scheduling phase is shown in Figure 6.

Step 1: after the server management center detects the migration command, it starts the construction of the VM migration group and builds the VM migration group G according to the service relevance as the load group that needs to be migrated. Then, send a VMM request [Req] to the VMM management center.

Step 2: after the idle server receives the VMM request information, it calculates its own failure rate, load capacity, and other information.

Step 3: the idle server returns the reply message [reply] to the original server.

Step 4: if the original server is adjacent to the idle server, that is, there is a TCP link, send a reply message [reply] directly to the original server; otherwise, establish a new link first, and then send [reply] to the server.

Step 5: the original server receives [reply], selects the appropriate target server according to the load information of the idle server, and prepares to enter the migration execution phase.

4.3. Implementation of the Migration Execution Phase. The migration execution phase mainly performs VM migration. The first step is to establish a temporary storage device to calculate the service priority according to formula (8). The second step is to directly send the original server to the target server and migrate the memory page to the TD through the running state of the VM. In the third step, the target server moves out the memory page from the TD and obtains the required memory page from the original server. The second and third steps are carried out at the same time and are collectively referred to as memory page push. Figure 7 shows the process of the migration execution phase.

Next, the specific implementation process of the migration execution phase is described in detail:

Step 1: the original server obtains the resource usage of each server from the VMM manager and obtains the candidate server list $H = S_1, S_2 \dots S_m$ of TD according to the result of the migration scheduling stage. Select some physical servers in the list as TD, which can be used by overloaded servers to quickly move out of VMs and move in memory pages to the target server.

```

Input: initial list push_tab
Output: push_tab
Method:
(1) Receive page faults request;
(2) Send pages to target server;
(3) push_tab.delete (pages);
(4) Service  $I_1 = \text{getService}(\text{page})$ ; //get service related to fault page.
(5) for each page in unMigrationPages do //unMigrationPages are pages unmigrated.
(6)   Service  $I_j = \text{getService}(\text{page})$ ;
(7)   if ( $I_j$  is related to  $I_1$ ) then
(8)     if ( $I_j$  replay on  $I_1$ ) then
(9)       push_tab[page_index] = push_tab[page_index] + ( $V_0 / \text{depended}(I_1)$ );
(10)    else
(11)      push_tab[page_index] = push_tab[page_index] +  $V_0$ ;
(12)    end if
(13)  else
(14)    push_tab[page_index] = push_tab[page_index] +  $\text{PR}((\text{pages}(I_j))/2)$ ;
(15)  end if
(16) end for
(17) return push_tab;
    
```

ALGORITHM 2: Page push algorithm based on service priority (PBSP).

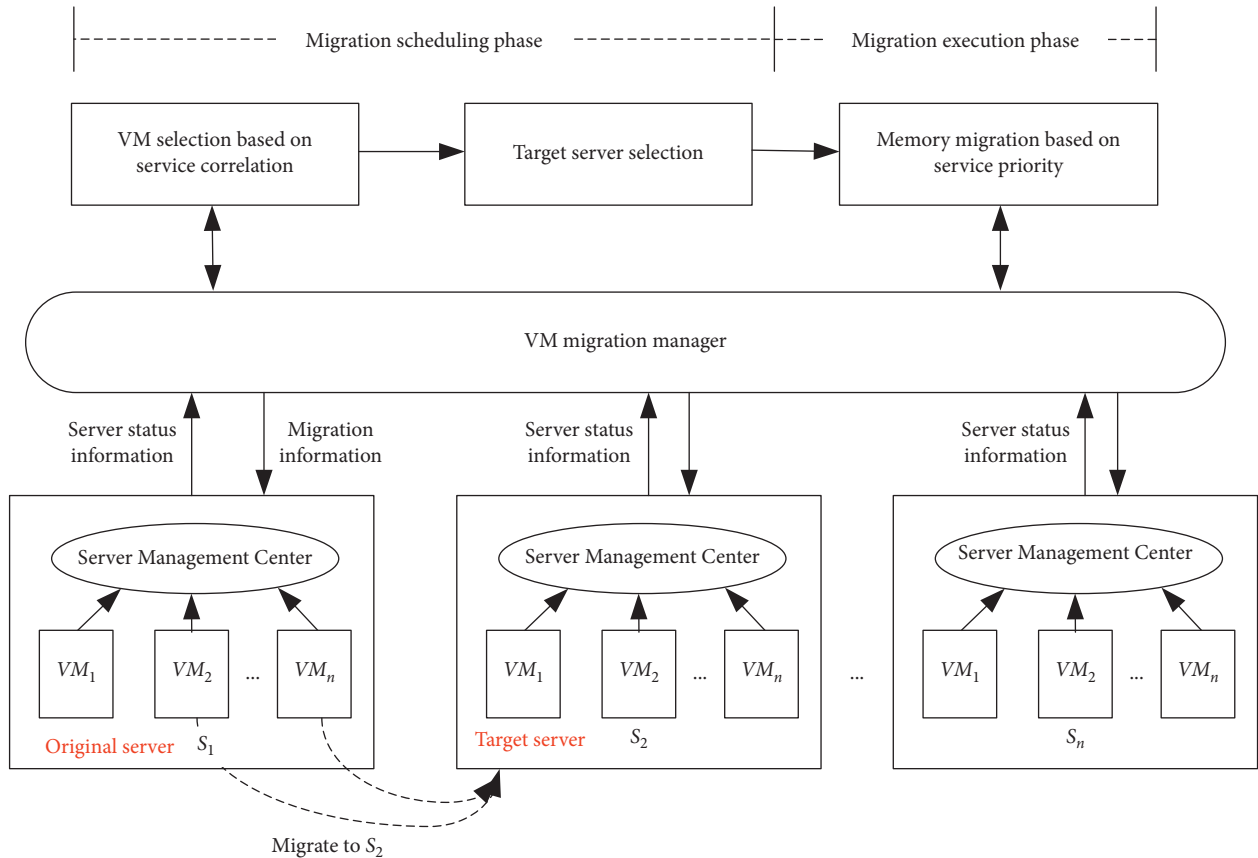


FIGURE 5: VM live migration framework.

Step 2: send the minimum execution conditions such as the execution status of each VM in the VM migration group directly to the target server and resume the execution of the VM on the target server.

Step 3: move the memory pages in the VM migration group into the TD in the order of service priority. If the VM migrated on the target server has a page fault interruption at this time, it directly requests the

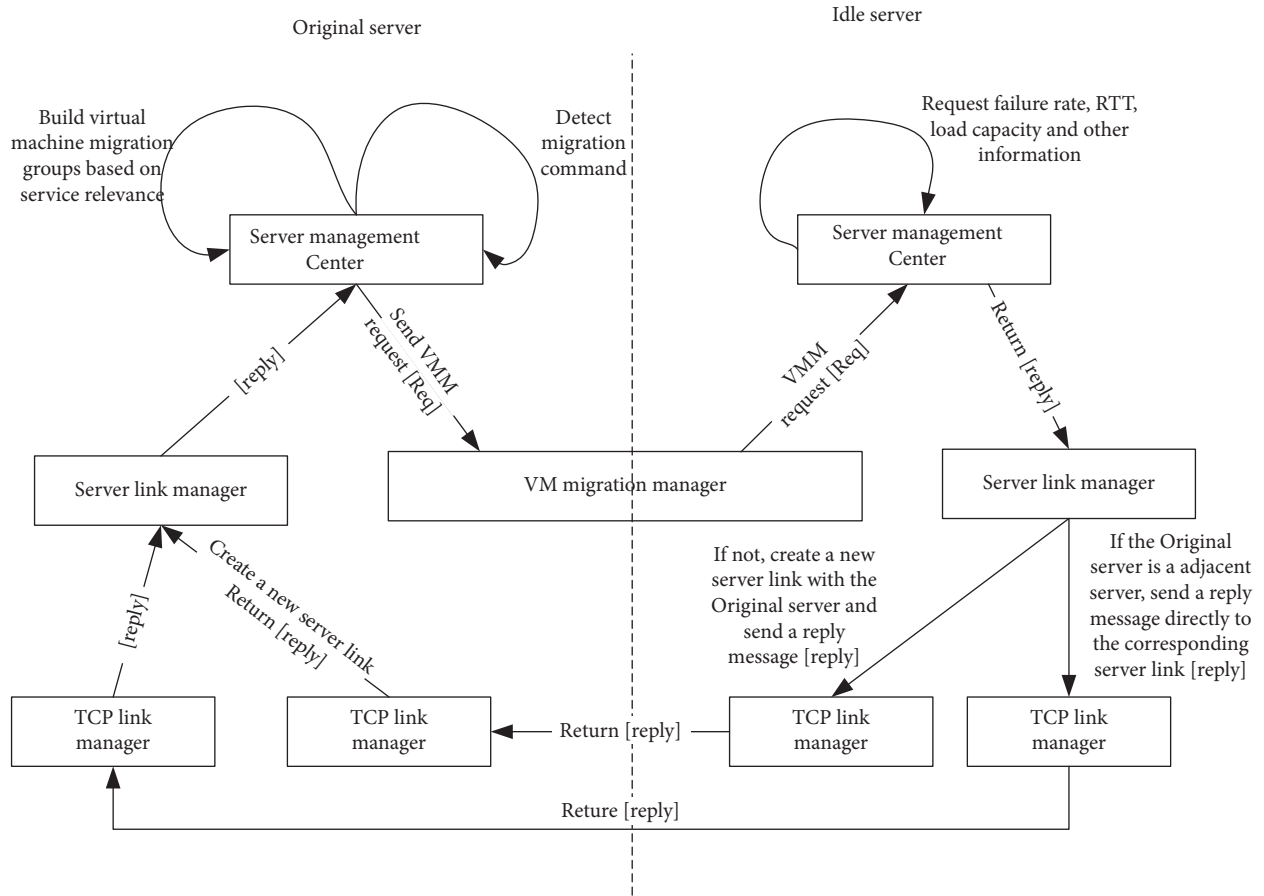


FIGURE 6: Migration scheduling stage.

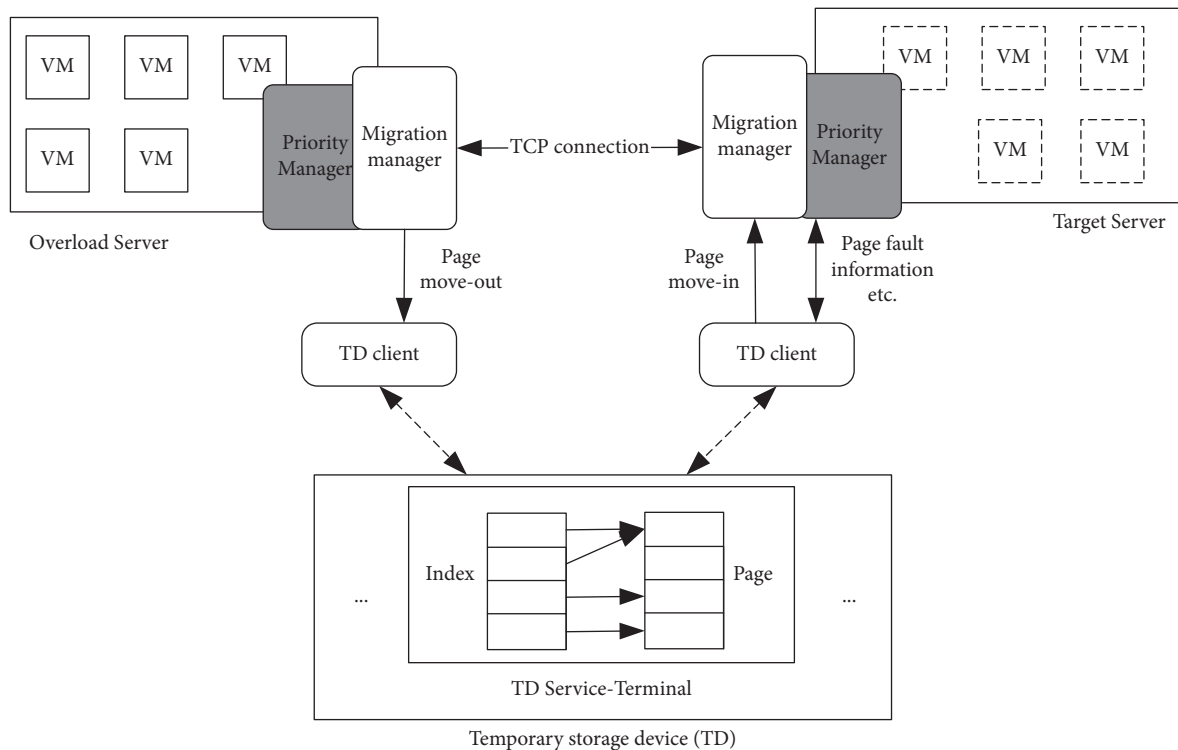


FIGURE 7: Migration execution phase.

memory page from the original server, and the original server sends the page fault to the target server and, at the same time, updates the order in which the VM memory page is pushed. Repeat this process until all the memory in the VM is sent to the TD.

Step 4: the target server migrates the memory pages in the TD to the server in the order of service priority. This process is synchronized with the third step. If a page fault occurs after the end of the third step, a memory page is directly requested from the TD. Repeat this process until the target server finishes receiving memory.

5. Results Analysis

5.1. Experimental Protocol. In the experiment, this paper uses a total of 6 PCs based on the actual situation, of which one is used as a VMM management node to deploy migration management strategies, and the other 5 are used as edge nodes. The configuration of each PC is shown in Table 1.

The Eucalyptus cloud computing platform is open source, easy to use, and rich in management interfaces [30]. This article uses the Eucalyptus architecture to build the above devices into a cluster system. In this experiment, the management node is set as the VM migration manager, and the VMs are run on edge nodes to simulate providing services to terminal devices. This article deploys the VM placement method and memory migration method on the system to implement the VM migration framework. The topology of the final cluster is shown in Figure 8.

All VMs in the system use KVM/QEMU 1.6.50, running Ubuntu as a suboperating system, with 2 virtual CPUs (vCPU), and each VM is configured with different memory sizes such as 512M, 1024M, 2G, etc.

In order to verify the effectiveness of the VM migration strategy based on service relevance, the experiment was divided into four groups, and the following questions were tested, respectively:

- (1) Whether the strategy can reduce the communication overhead between servers?
- (2) Whether the strategy can load balance quickly?
- (3) Whether the memory migration method of this policy is efficient?

5.2. Analysis of Results

Experiment 1. This experiment is to verify the effectiveness of the VM placement method based on service relevance (SRVMP) and simulate the round-robin scheduling (RR) algorithm [31], the least connection (LC) algorithm [32], and the ant colony (AG) algorithm [33]. In order to compare the communication overhead of each algorithm, this paper divides each algorithm into single VM migration (single) and multiple VM migration (multiple). The experimental comparison chart is shown in Figure 9. It can be seen from Figure 9 that the communication overhead of the SRVMP

algorithm is much smaller than that of the RR algorithm and the LC algorithm, and the communication overhead of multivirtual machine RR algorithm is lower than single-virtual machine RR algorithm, and multivirtual machine LC algorithm has lower communication overhead than single-virtual machine LC algorithm. This is because multivirtual machine migration algorithm migrates multiple VMs to the same server, which avoids the communication overhead caused by VMs being scattered to different servers. The SRVMP algorithm considers the service relevance between VMs during multivirtual machine migration and groups closely related VMs into a migration group. Therefore, the SRVMP algorithm designed in this paper can reduce the communication overhead between servers during migration.

Experiment 2. Experiment 2 needs to test the load balancing ability and task execution time of the system, and the combined result of the two is used as a measure of the effectiveness of load balancing.

The load balancing capability is measured by the load balance degree. The load balance degree refers to the standard deviation between the server resource utilization and the average data center resource utilization, which reflects the system load distribution. And the smaller the degree is, the more balanced the load is. The calculation formula is as follows:

$$L_{\text{factor}} = \sqrt{\frac{\sum_{i=1}^n (L_i - \bar{L}_{\text{datacenter}})^2}{n}}, \quad (12)$$

where L_{factor} represents the load balance degree, n represents the total number of servers in the data center, L_i represents the resource utilization of the servers S_i , and $\bar{L}_{\text{datacenter}}$ denotes the average utilization of the data center resources.

The task execution time indicates the duration from entering the processing queue to completing the task processing. This paper mainly takes the form of randomly generating multiple tasks, such as generating the number of $10^3 - 5 * 10^3$ tasks at regular intervals and then recording the results of several time units. In the task processing time experiment, the efficiency of different algorithms is verified by different task numbers. Besides, so as to avoid accidental factors, the average is taken as the final result through 30 experiments.

The algorithms in this experiment are all based on multivirtual machine migration, and on the basis of Experiment 1, a self-heuristic algorithm ant colony (AG) algorithm is added for comparison. The results of the experiment are shown in Figure 10.

According to Figure 10(a), we can see that the SRVMP algorithm is relatively evenly distributed in the CPU load balance degree. Although it is lower than the AG algorithm, the load balancing degree is roughly distributed around 10. The reason for this phenomenon is that this paper uses the greedy strategy to chooses the server with the least migration cost as the target server, so the communication cost and other indicators are not optimal. Comparing the RR algorithm and the LC algorithm, the algorithm in this paper has a greater advantage in load balance degree. Similarly,

TABLE 1: Experimental configuration.

	Management node	Other nodes
CPU	2.3 GHz	2.3 GHz
RAM	32 GB	16 GB
Hard disk	1 TB	750 GB
Bandwidth	1 Gbps	1 Gbps

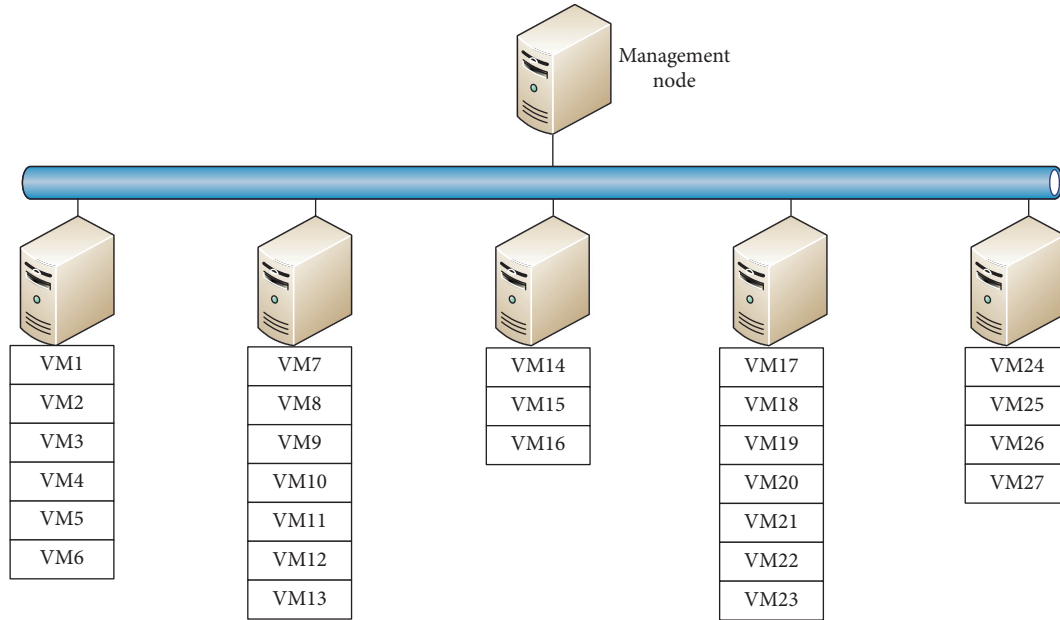


FIGURE 8: Experimental environment topology.

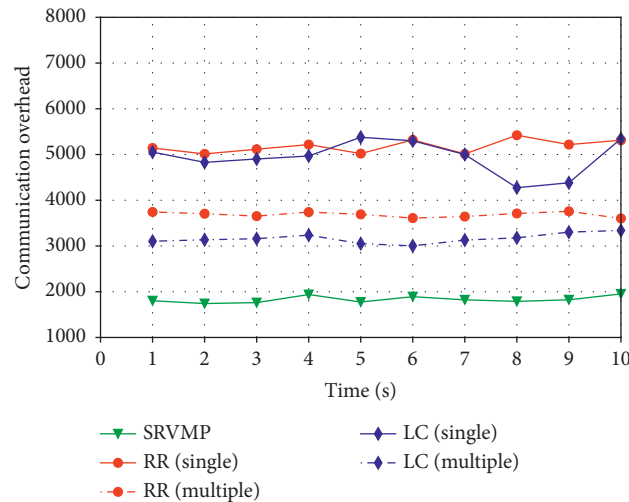


FIGURE 9: Comparison of communication overhead.

Figures 10(b) and 10(c) also prove that the SRVMP algorithm can achieve system load balancing. Since the AG algorithm has a higher balance degree, this paper will do an experiment to check the execution time of the task. The experimental result is shown in Figure 11.

It can be seen from Figure 11 that the SRVMP algorithm proposed in this paper has a greater advantage in task

execution time. When the number of tasks is small, there is not much difference in the execution time, but when the number of tasks gradually increases, the execution time of the RR algorithm and the LC algorithm increases significantly. The time complexity of the SRVMP algorithm is $O(n^2)$, the time complexity of the ant colony algorithm is $O(nc \cdot n^3)$, and nc is the number of iterations of the ant

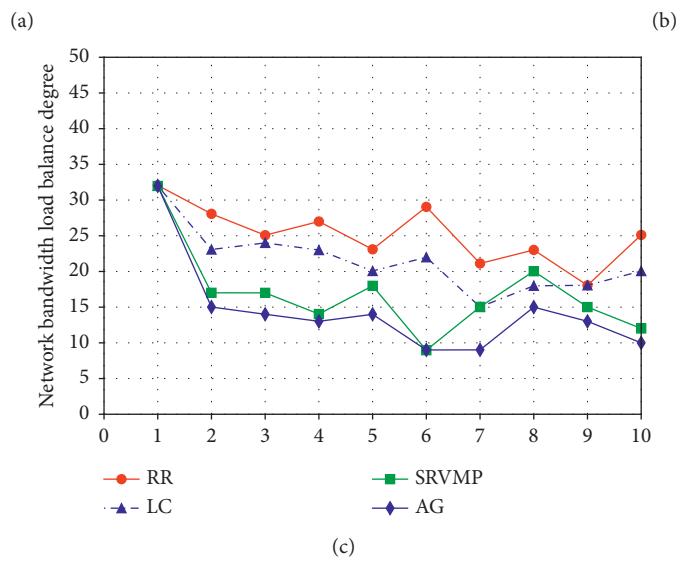
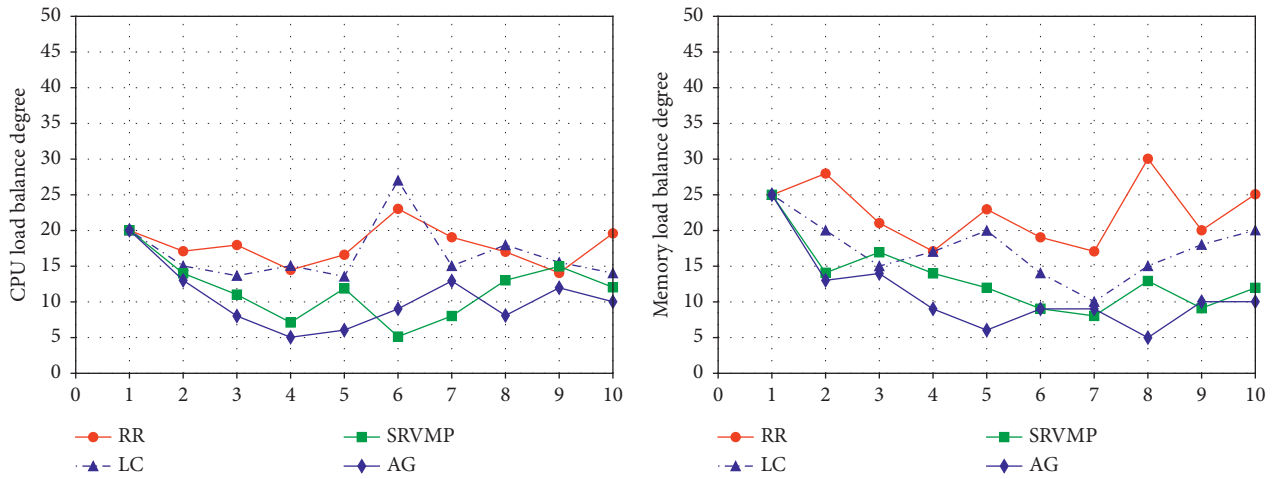


FIGURE 10: Comparison of load balance degree.

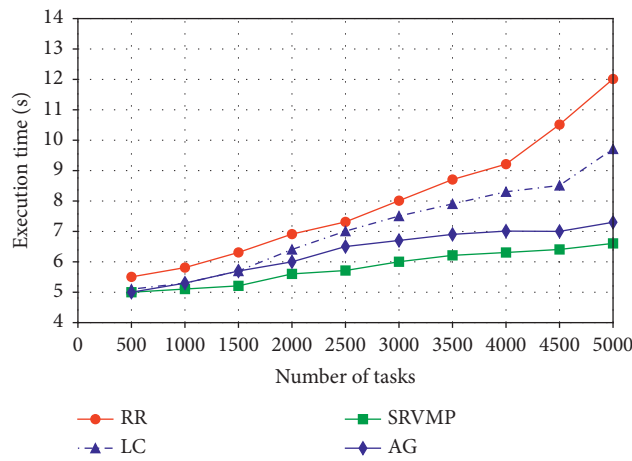


FIGURE 11: Comparison of task execution time.

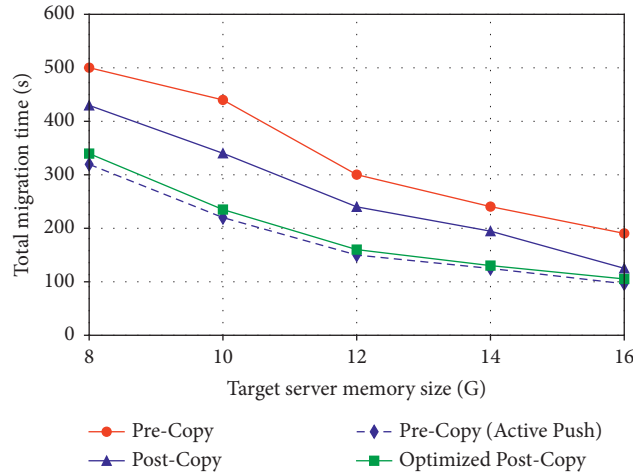


FIGURE 12: Comparison of total migration time (TMT).

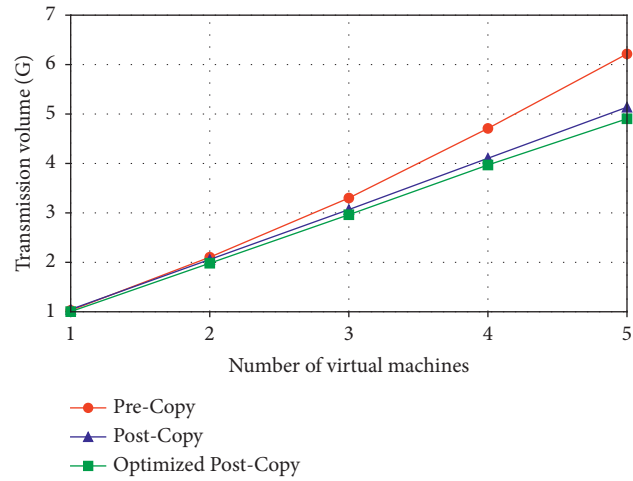


FIGURE 13: Comparison of total transferred data.

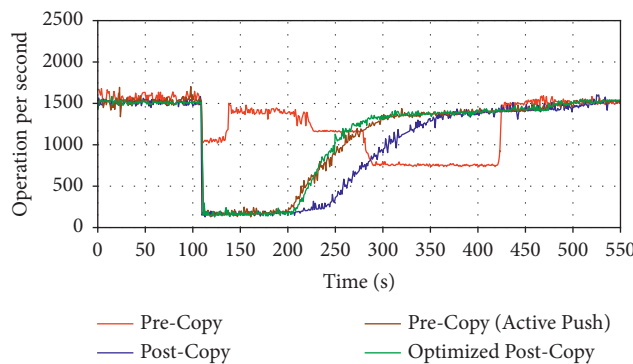


FIGURE 14: Comparison of operations per second.

colony algorithm. The AG algorithm requires multiple iterations. Therefore, although the SRVMP algorithm is slightly worse than the AG algorithm in system load balancing, it does not affect the task execution speed.

Experiment 3. The VM migration strategy based on service relevance designed in this paper uses an improved post-copy method (optimized post-copy) PCBSP to perform memory migration.

This experiment compares PCBSP with pre-copy and post-copy and uses total migration time (TMT), total migrated data, and page fault conditions as the criteria for verifying the efficiency of the method [34].

5.2.1. Total Migration Time (TMT). We set the target server's memory in the form of virtual memory in the form of 8G, 10G, 12G, 14G, and 16G and checked the changes of TMT in the four methods. The experimental results are shown in Figure 12.

Figure 12 shows that as the target server memory gets larger, the memory pressure gets smaller and smaller, and the TMT gap of each method gradually decreases. As can be seen from the figure, pre-copy takes the longest time, post-copy takes slightly shorter time, and post-copy (Active Push) further shortens the total time, because post-copy (Active Push) combines active push of memory and on-demand requests for memory to speed up the migration speed. The optimized post-copy method used in this article takes slightly longer than post-copy (Active Push). This is because the optimized post-copy method needs to first migrate the memory pages to TD and then to the target server.

5.2.2. Total Migrated Data. In this experiment, the original server and the target server were kept unchanged, and we compared the changes in the amount of migrated data, respectively, when the number of VMs is 1, 2, 3, 4, and 5. Since post-copy and post-copy (Active Push) have roughly the same amount of migrated data, the experiment compared the amount of migrated data by pre-copy, post-copy, and optimized post-copy methods. Figure 13 shows the comparison of the transferred data volume of the three methods. In the figure, the pre-copy transferred data volume is the largest, because the pre-copy method needs to repeatedly transfer the dirty pages synchronously, while the post-copy method only needs to migrate the memory page once. The optimized post-copy method takes into account the problem of data redundancy, so the amount of transferred data is the least, reducing the migration overhead, and effectively improving the efficiency of VM migration.

5.2.3. Page Fault Interruption. In order to show the trade-off between MT and migration performance, select a suitable VM to be migrated as the database server, query the database content through Yahoo Cloud Servicing Benchmark (YCSB) [35], and calculate each data based on the data calculated by YCSB. The number of operations per second is used to measure migration performance. This paper shows the changes in the number of operations per second of several methods over time. According to continuous monitoring of migration performance, the effect is shown in Figure 14.

The figure shows that, from about 110 seconds, the post-copy-related methods have always remained at a low level from the beginning of the migration. This is caused by a large number of page faults. At this time, the request for page faults and the target server receives memory pages to

compete for traffic. The performance of the pre-copy method will also decline, but the decline is relatively gentle, because the pre-copy method does not have the effect of page faults. We compared it with the other two post-copy methods; the method in this paper recovers faster. Although the performance of the post-copy method decreases less, the MT is the longest. We consider that the application background of this paper is to complete the VMM as quickly as possible, so the trade-off between MT and migration performance should choose MT as a more important performance indicator when the migration performance is acceptable.

In summary, the post-copy memory migration method based on service priority proposed in this chapter can indeed reduce the VM migration time and reduce the occurrence of page faults when the migration performance is less affected; that is, it can quickly and efficiently complete VM migration.

6. Conclusion

Edge computing-based IoT provides a platform for data calculation and storage for the IoT devices. The data generated by IoT devices contains a large amount of user identity information, location information, and sensitive information. Therefore, data security and privacy issues in edge computing-based IoT are becoming increasingly prominent. Side-channel attacks steal the private information of other virtual machines by coresident virtual machines to bring huge security threats to edge computing. Virtual machine migration is the main way to defend against side-channel attacks. When selecting VM for migration, the degree of association between services is not considered. If some closely connected VMs are migrated to different servers, it will bring a lot of communication overhead during the calculation and migration process. In the process of VM memory migration, the pre-copy method is often used, which needs to repeatedly transfer a large number of dirty pages, thereby increasing the migration time of the virtual machine.

For the above problems, this paper proposes a VM migration strategy based on service relevance. First, define the service relevance factor to quantify the degree of relevance between services. Then, design a VM selection strategy based on service relevance, and group closely related VMs into a VM migration group to migrate. Finally, we design a post-copy memory migration method based on service priority (PCBSP); it can quickly migrate out of the VM migration group and effectively reduce the page fault rate. Through comparative experiments, it is verified that the VM selection method based on service relevance can effectively reduce the communication overhead in VM live migration. However, there are still two limitations in this paper. The next step of this paper is as follows: (1) in the VM migration, the downtime caused by server failure or overload is not considered. Once failure or downtime occurs, the VM migration will be invalid, and it is difficult to recover the VM after failure. The next step should consider how to deal with failure or downtime when using this method to migrate VMs. (2) PCBSP only calculates the service priority uniformly according to the dependency and dependent

relationship of the service, without considering the specific situation. In the next step, we should consider the specific situation to make the calculation of service priority more accurate and further reduce the page fault rate.

Data Availability

The experimental data of this study are available from the corresponding author upon reasonable request.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this article.

Acknowledgments

This work was supported in part by the Major Projects of Natural Science Research in Universities in Jiangsu Province under Grant 17KJA413001.

References

- [1] X.-J. Chen, B.-D. Chen, X.-M. Jiang, X.-B. Chen, and W.-H. Cai, "Improved cloud computing architecture for the Internet of Things," *Journal of Internet Technology*, vol. 17, no. 4, pp. 683–693, 2016.
- [2] W. Yu, F. Liang, X. He, W. G. Hatcher, C. Lu, and X. Yang, "A survey on the edge computing for the Internet of Things," *IEEE access*, vol. 6, pp. 6900–6919, 2017.
- [3] B. Omoniwa, R. Hussain, M. A. Javed, S. H. Bouk, and S. A. Malik, "Fog/edge computing-based IoT (FECIoT): architecture, applications, and research issues," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 4118–4149, 2018.
- [4] M. S. Sheikh, J. Liang, and W. Wang, "Security and privacy in vehicular ad hoc network and vehicle cloud computing: a survey," *Wireless Communications and Mobile Computing*, vol. 2020, Article ID 5129620, 25 pages, 2020.
- [5] C. Su and Q. Zeng, "Survey of CPU cache-based side-channel attacks: systematic analysis, security models, and countermeasures," *Security and Communication Networks*, vol. 2021, Article ID 5559552, 15 pages, 2021.
- [6] 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*, L, USA, November 2009.
- [7] M. S. Inci, B. Gulmezoglu, G. Irazoqui, T. Eisenbarth, and B. Sunar, "Cache attacks enable bulk key recovery on the cloud," in *Proceedings of the International Conference on Cryptographic Hardware And Embedded Systems*, Santa Barbara, CA, August 2016.
- [8] Y. Zhang, Y. Mao, M. Xu, F. Xu, and S. Zhong, "Towards Thwarting Template Side-Channel Attacks in Secure Cloud Deduplications," *IEEE Transactions on Dependable and Secure Computing*, vol. 18, no. 3, pp. 1008–1018, 2019.
- [9] D. Park, G. Kim, D. Heo, S. Kim, H. Kim, and S. Hong, "Single trace side-channel attack on key reconciliation in quantum key distribution system and its efficient countermeasures," *ICT Express*, vol. 7, no. 1, pp. 36–40, 2021.
- [10] C. Yang, Y.-f. Guo, H.-c. Hu, Y.-w. Wang, Q. Tong, and L.-s. Li, "Driftor: mitigating cloud-based side-channel attacks by switching and migrating multi-executor virtual machines," *Frontiers of Information Technology & Electronic Engineering*, vol. 20, no. 5, pp. 731–748, 2019.
- [11] X. Wang, L. Wang, F. Miao, and J. Yang, "SVMDF: a secure virtual machine deployment framework to mitigate co-resident threat in cloud," in *Proceedings of the 2019 IEEE Symposium on Computers and Communications (ISCC)*, Barcelona, Spain, June 2019.
- [12] S. B. Melhem, A. Agarwal, N. Goel, and M. Zaman, "A Markov-based prediction model for host load detection in live VM migration," in *Proceedings of the 2017 IEEE 5th International Conference on Future Internet of Things and Cloud (FiCloud)*, Prague, Czech Republic, August 2017.
- [13] S. Sotiriadis, N. Bessis, and R. Buyya, "Self managed virtual machine scheduling in cloud systems," *Information Sciences*, vol. 433-434, pp. 381–400, 2018.
- [14] L. Liu, S. Zheng, H. Yu, V. Anand, and D. Xu, "Correlation-based virtual machine migration in dynamic cloud environments," *Photonic Network Communications*, vol. 31, no. 2, pp. 206–216, 2016.
- [15] M. Rajabzadeh and A. T. Haghghat, "Energy-aware framework with Markov chain-based parallel simulated annealing algorithm for dynamic management of virtual machines in cloud data centers," *The Journal of Supercomputing*, vol. 73, no. 5, pp. 2001–2017, 2017.
- [16] J. Xu and J. A. Fortes, "Multi-objective virtual machine placement in virtualized data center environments," in *Proceedings of the 2010 IEEE/ACM Int'l Conference on Green Computing and Communications & Int'l Conference on Cyber, Physical and Social Computing*, Hangzhou, China, December 2010.
- [17] A. Verma, P. Ahuja, and A. Neogi, "pMapper: power and migration cost aware application placement in virtualized systems," in *Proceedings of the ACM/IFIP/USENIX International Conference on Distributed Systems Platforms and Open Distributed Processing*, Heidelberg, Germany, December 2008.
- [18] M. K. Gupta and T. Amgoth, "Resource-aware virtual machine placement algorithm for IaaS cloud," *The Journal of Supercomputing*, vol. 74, no. 1, pp. 122–140, 2018.
- [19] R. Kanniga Devi, G. Murugaboopathi, and M. Muthukannan, "Load monitoring and system-traffic-aware live VM migration-based load balancing in cloud data center using graph theoretic solutions," *Cluster Computing*, vol. 21, no. 3, pp. 1623–1638, 2018.
- [20] L. Lin, J. Chen, P. K. Kudjo, and O. Michael, "A novel routing protocol for content-based publish/subscribe model in mobile sensor networks," *International Journal of Distributed Sensor Networks*, vol. 15, no. 9, 2019.
- [21] F. Zhang, G. Liu, X. Fu, and R. Yahyapour, "A survey on virtual machine migration: challenges, techniques, and open issues," *IEEE Communications Surveys & Tutorials*, vol. 20, no. 2, pp. 1206–1243, 2018.
- [22] U. Mandal, P. Chowdhury, M. Tornatore, C. U. Martel, and B. Mukherjee, "Bandwidth provisioning for virtual machine migration in cloud: strategy and application," *IEEE Transactions on Cloud Computing*, vol. 6, no. 4, pp. 967–976, 2016.
- [23] M. R. Desai and H. B. Patel, "Efficient virtual machine migration in cloud computing," in *Proceedings of the 2015 15th International Conference On Communication Systems And Network Technologies*, Gwalior, India, April 2015.
- [24] J. Zhang, F. Ren, R. Shu, T. Huang, and Y. Liu, "Guaranteeing delay of live virtual machine migration by determining and provisioning appropriate bandwidth," *IEEE Transactions on Computers*, vol. 65, no. 9, pp. 2910–2917, 2015.

- [25] S. Nathan, U. Bellur, and P. Kulkarni, "On selecting the right optimizations for virtual machine migration," *ACM SIGPLAN Notices*, vol. 51, no. 7, pp. 37–49, 2016.
- [26] K. Su, W. Chen, G. Li, and Z. Wang, "Rpff: a remote page-fault filter for post-copy live migration," in *Proceedings of the 2015 IEEE International Conference on Smart City/SocialCom/SustainCom (SmartCity)*, Chengdu, China, December 2015.
- [27] G. Sun, D. Liao, V. Anand, D. Zhao, and H. Yu, "A new technique for efficient live migration of multiple virtual machines," *Future Generation Computer Systems*, vol. 55, pp. 74–86, 2016.
- [28] Z. Lei, E. Sun, S. Chen, J. Wu, and W. Shen, "A novel hybrid-copy algorithm for live migration of virtual machine," *Future Internet*, vol. 9, no. 3, p. 37, 2017.
- [29] U. Deshpande, D. Chan, S. Chan, K. Gopalan, and N. Bila, "Scatter-gather live migration of virtual machines," *IEEE Transactions on Cloud Computing*, vol. 6, no. 1, pp. 196–208, 2015.
- [30] S. Yadav, "Comparative study on open source software for cloud computing platform: Eucalyptus, openstack and opennebula," *International Journal of Engineering Science*, vol. 3, no. 10, pp. 51–54, 2013.
- [31] E. L. Hahne, "Round-robin scheduling for max-min fairness in data networks," *IEEE Journal on Selected Areas in Communications*, vol. 9, no. 7, pp. 1024–1039, 1991.
- [32] X. Ren, R. Lin, and H. Zou, "A dynamic load balancing strategy for cloud computing Platform based on exponential smoothing forecast," in *Proceedings of the 2011 IEEE International Conference on Cloud Computing and Intelligence Systems*, Beijing, China, September 2011.
- [33] M. A. Tawfeek, A. El-Sisi, A. E. Keshk, and F. A. Torkey, "Cloud task scheduling based on ant colony optimization," in *Proceedings of the 2013 8th International Conference on Computer Engineering & Systems (ICCES)*, Cairo, Egypt, November 2013.
- [34] D. Malhotra, "A critical survey of virtual machine migration techniques in cloud computing," in *Proceedings of the 2018 First International Conference On Secure Cyber Computing And Communication (ICSCCC)*, Jalandhar, India, December 2018.
- [35] V. Abramova, J. Bernardino, and P. Furtado, "Evaluating cassandra scalability with YCSB," in *Proceedings of the International Conference On Database And Expert Systems Applications*, Munich, Germany, September 2014.

Research Article

Combinatorial Spectrum E-Auction for 5G Heterogeneous Networks: A Zether-Based Approach

Zijun Zhao ¹, Zuobin Ying,^{2,3} Zhiming Cai ³ and Jianfeng Ma¹

¹School of Physical & Information Technology, Anhui University, Hefei 230601, China

²School of Computer Science & Technology, Anhui University, Hefei 230601, China

³Faculty of Data Science, City University of Macau, Taipa 999078, Macau

Correspondence should be addressed to Zhiming Cai; caizhiming@cityu.mo

Received 10 September 2021; Revised 14 October 2021; Accepted 23 October 2021; Published 16 November 2021

Academic Editor: Ke Gu

Copyright © 2021 Zijun Zhao et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

5G heterogeneous network (HetNet) is a novel network topology that integrates various kinds of wireless access technologies such as 4G Long-Term Evolution (LTE), Wi-Fi, and so on. Despite greatly improving spectrum efficiency, it poses enormous challenges to spectrum e-auction. Firstly, due to high mobility, bidders may be interested in different spectrums in terms of time or geolocation. Secondly, one's bidding value should be protected against rival bidders or adversaries to avoid vicious competition as well as privacy leakage. Thirdly, the ubiquitous HetNet requires a trustworthy distributed auction framework rather than a centralized auctioneer-based pattern. Aiming at overcoming these obstacles above, we proposed a blockchain-based combinatorial spectrum e-auction framework. Different from other blockchain-based solutions of using SGX to realize trust processing in the auction phase, we adopt Zether, a privacy-preserving smart contract, as the main building block. Besides, the bidding value is preserved from the beginning to the end, even though the time-consuming Paillier homomorphic encryption and garbled circuits are absent. We provide the auction security by leveraging Σ -Bullets, a zero-knowledge proof mechanism. Theoretical analysis and extensive evaluation also indicate that our approach is better than the state-of-the-art works in terms of efficiency and effectiveness.

1. Introduction

The fifth-generation (5G) mobile network is expected to promote the connection of everything that demands a low-latency Internet connection, from IoT devices and appliances to self-driving cars, paving the way for an environment where every device is smart and connected. According to the forecast released from Cisco, the overall mobile data traffic is expected to grow to 77 exabytes per month by 2022, a seven-fold increase over 2017 [1]. As a result, the existing wireless network capacity has been unable to support the explosive growth of data traffic and the ubiquitous demand for high-quality communication. New wireless and network technologies are demanded to solve the contradiction between the limited wireless bandwidth resources of the existing network and a large number of high-speed transmission requirements. Heterogeneous network (HetNet), emerging

as a novel network topology which integrates various kinds of wireless access technologies (e.g., 5G, 4G Long-Term Evolution (LTE), Wi-Fi, Universal Mobile Telecommunications System (UMTS), and so on), is deemed to be the most promising solution against the above challenges. However, the scarce spectrum resource has become an obstacle in HetNet deployment. Considering the wide coverage of 5G services, 3rd Generation Partnership Project (3GPP) introduces the idea of unlicensed 5G in Release 16, which is expected to solve the problem of 5G spectrum scarcity [2]. Nevertheless, a field test released by Aviat Networks points out that the use of unlicensed equipment in the 6 GHz frequency band will affect the microwave point-to-point links operating on this frequency band and cause interference to existing users [3]. Obviously, the lack of coordination between these unlicensed frequency band applications will directly lead to interference between

different services and lead to a series of other unfavorable consequences due to no spectrum coordination. Therefore, the effective allocation of the 5G spectrum has become a key factor that affects the availability of HetNet.

Spectrum e-auction, which is considered to be one of the most effective ways of solving the spectrum allocation problem, has been widely researched in the past few years [4–7]. Recently, on account of the rise of the smart contract, some blockchain-based frameworks have also been constructed [8–11]. There are three main entities participant in the spectrum e-auction process, namely, seller(s), auctioneers, and buyer(s). The most common workflow of a spectrum e-auction is as follows. (1) Seller (maybe more than one seller, *e.g.*, double auction) releases the spectrum resources to the auctioneers. (2) Buyers (maybe only one buyer, *e.g.*, reverse auction) submit their bid values as well as other information (*e.g.*, location and account address) to the auctioneer. (3) Auctioneer judges the winner according to the bid value and then returns the result to both the seller and the bidders. (4) Auctioneer refunds the bids to those buyers who have not won in the auction. Seller and winner buyers finish the deal. This workflow could be implemented to most of the spectrum auction schemes. However, the actual application scenarios of HetNet put forward some specific requirements for spectrum auctions. We summarize the most challenging issues as follows:

- (i) In HetNet, the buyer may be interested in more than one spectrum. Besides, buyer (*e.g.*, the autonomous vehicle) may not be fixed in one location but occupies the spectrum resources in a certain location within a certain period of time. That is to say, the buyer might be interested in a bundle of the spectrum related to both geolocation and time. Thus, the corresponding combinatorial auction mode has to be considered.
- (ii) The buyer's bidding value should be protected against the other rival bidders or adversaries. Existing spectrum auction schemes rely either on some expensive cryptographic tools (*e.g.*, garbled circuits (GC) and homomorphic encryption) or on the implementation of trusted processors (*e.g.*, Intel SGX). These approaches would not only increase time consumption but also have to make more hypothesis.
- (iii) Most of the existing sealed-bid e-auction schemes require a trusted third-party auctioneer to ensure the fairness of bidding. Alternatively, assuming that the auctioneer is semihonest, then an additional semihonest auction agent is also needed under the restriction that it would not collude with the auctioneer. A fully decentralized spectrum auction scheme without trusted third party has not been effectively constructed.

Motivated by solving the aforementioned issues simultaneously, we proposed a combinatorial spectrum e-auction for 5G HetNet by leveraging the latest privacy-preserving smart contract theory named Zether [12]. Our ultimate goal

is to design a combinatorial e-auction scheme which considers both bidding value privacy and practicality. Hereby, we summarize our contributions:

- (i) As far as we know, we are the unique to construct a combinatorial e-auction scheme based on Zether. We not only give the concrete construction but also design the auction procedure on the Zether smart contract. It is worth noting that our scheme could be easily extended to other account-based blockchain platforms (*e.g.*, Hyperledger Fabric, EOS, and so on).
- (ii) The bidding value of the buyer is protected without introducing the expensive cryptographic tools or trusted processors. We take advantage of the additive homomorphic feature of ElGamal encryption, thereby reducing the complexity of the entire scheme. Technically speaking, this is the first blockchain-based e-auction without a trusted third party.
- (iii) To ensure the correctness of the encrypted transactions, the zero-knowledge proof (ZK proof) has to be included in the smart contract, which is also the expensive part in most of the blockchain-based auction schemes. The experimental results indicate that our proposed scheme is superior than state-of-the-art works in terms of time and gas consumption.

2. Related Work

The rapid development of 5G communication as well as the new architecture of HetNet facilitates the speed and diversity of accessing the Internet. Nonetheless, the high density of network infrastructure and the mobile nodes aggravate the scarceness of the spectrum. Spectrum e-auction looks prophetic against this dilemma. Since the auction procedure can be regarded as a multi-player game among different bidders, the bid value needs to be protected to avoid malicious competitions or collude attacks. Miao Pan et al. proposed a secure spectrum auction scheme to prevent the frauds of the insincere auctioneers between the auctioneer and the bidders by utilizing the Paillier cryptosystem, namely, *THEMIS* [4]. Wang et al. extended the security concerns to geolocation and time dynamics other than bid value only by introducing *PROST* [5]. However, to achieve the design goals, *PROST* uses a series of expensive cryptographic tools such as Paillier homomorphic encryption, oblivious transfer, and garbled circuits to construct the atomic blocks for the secure auction protocol. Afterwards, *ARMOR* [6] and *PS-TAHES* [7] are proposed to tackle the security issues in the heterogeneous spectrum, respectively. Both of these works leverage Paillier homomorphic encryption and garbled circuits along with some other cryptographic tools, and the difference is that *ARMOR* concentrates on combinatorial auction, while *PS-TAHES* focuses on double auction. Cheng et al. put forward another lightweight auction framework without using Paillier algorithm, namely, *SLISA* [13]. A set of subprotocols is designed by integrating additive secret sharing and garbled

circuits. *SLISA* provides strong security guarantees related to the bidders in the double auction.

In the past few years, blockchain technology has attracted tremendous attention. As a distributed ledger with the inherent temper-resistant feature, the new paradigm of “blockchain + x (i.e., everything)” reaches a consensus that it could revolutionize every aspect of our lives. The subsequent deployment of smart contract in blockchain 2.0 (i.e., Ethereum) makes it more practical for financial applications. Weiss et al. proposed an idea of spectrum management via adopting blockchain. They widely examined the blockchain application in spectrum sharing and, in the meantime, specified that a number of areas would benefit from further research [15]. Thereafter, considering the spectrum shortage dilemma, Zhou et al. put forward a blockchain-based secure spectrum sharing scheme for 5G HetNet, in which the underutilized spectrum allocated to the human-to-human (H2H) users could be shared with the machine-to-machine (M2M) communications. However, security claimed in this work is just the security guaranteed by the blockchain itself. Auction between the primary user and secondary user via the smart contract is totally transparent to everyone [10]. Wu et al. first considered the collusion coalitions among selfish auction participants and constructed a decentralized collusion-resistant e-auction system on Ethereum, named *CREAM* [8]. However, since the transactions on Ethereum are public, to protect bid privacy, *CREAM* designs a two-phase bidding process, *commitBid* and *revealBid*. After bid commitment, all bidders still have to trigger the *revealBid* to launch the auction algorithm. That is to say, rival bidders could still observe the true bid of a bidder. To eliminate the hypothetical trusted auctioneer (in some studies, if the auctioneer is semitrusted, then a semitrusted auction agent would be introduced in the premises that they would not collude with each other), some Software Guard Extension (SGX) approaches were proposed [9]. Recently, Chen et al. proposed *SAFE*, a general secure e-auction framework with privacy preservation [11]. It should be noted that this framework considered all the single-round (single-round auction stands for the bidders that can only submit their bids once) auction formats. Despite the fact that *SAFE* also leverages a bundle of cryptographic tools as well as the SGX, it is certainly one of the best spectrum e-auction approaches in state-of-the-art works. Moving one step forward, in this paper, we build a combinatorial spectrum e-auction scheme with privacy preserving based on Zether. We abandon the use of Paillier homomorphic encryption, garbled circuits, and SGX. Besides, we also reduce the gas consumption in the market cleanup phase. Finally, for ease of reading, we put the feature comparison in Table 1.

3. Preliminaries

3.1. Auction Terminologies. Here we present some auction terminologies used in our schemes.

- (i) *Sealed-Bid Auction.* A sealed-bid auction is an auction process in which all bidders submit sealed bids to the auctioneer at the same time so that no bidder knows the bids of other auction participants.

The sealed bid will not be opened before the specified date. The person with the highest bid is usually declared the winner of the bidding process.

- (ii) *Vickrey Auction.* Vickrey auction is also known as the *second-price sealed-bid auction*. All bids are sealed and sent to an auctioneer who can open all the bids. The highest bidder wins but only needs to pay the second-highest bid. It is highly centralized and does not protect the privacy of the bids [16]. If the bidder is interested in multiple items, Vickrey auction can be generalized to Vickrey–Clarke–Groves (VCG) auction.
- (iii) *Combinatorial Auction.* A combinatorial auction is a type of smart market in which participants can place bids on combinations of discrete heterogeneous items, rather than individual items or continuous quantities [17].

3.2. Zether and Zether Smart Contract (ZSC) [12]. Zether is a completely decentralized and confidential payment mechanism, compatible with Ethereum and other smart contract (SC) platforms. The fundamental of Zether is to realize transaction privacy via the smart contract, that is, hide the transaction amount and the balance of the account. For this purpose, mechanisms such as ElGamal encryption, pending transfer, and rolling over are designed. The payment mechanism is similar to Ethereum, which contains setup, user algorithms, and a smart contract. User algorithms, which contain seven subroutines, specify how users interact with Zether Smart Contract (ZSC). *CreateAddress* and *CreateBurnTx* check the input public keys to make sure that each pending transfer is rolled over. *CreateBurnTx* utilizes *ReadBalance* to recover the ZTH (ZTH is the confidential token of the Zether; the value of ZTH is related to the corresponding platform; for example, if the platform is Ethereum, then 1 ZTH = 1 ETH) from the account. *CreateFundTx* is utilized to deposit amounts to an account and *CreateTransferTX* is utilized to transfer money between one account and another. If the user wants to lock her account to an Ethereum address, she can use *CreateLockTx*. Otherwise, she can choose *CreateUnlockTx* to unlock an account.

ZSC has five methods: Fund, Burn, Transfer, Lock, and Unlock. Before executing the user algorithms, these functions would initiate the checkup process, such as checking the nonce or verifying a proof. If any of the checks does not succeed, the method outputs 0. Besides, ZSC also introduces a time horizon named *epoch*. The epoch length is denoted as E , and $E \geq 1$. A block’s epoch number at height h is described as $\lceil h/E \rceil$. In order to ensure the correctness, ZSC stipulates that a transaction should be processed in the same *epoch* as it is generated.

3.3. ElGamal Encryption. ElGamal encryption is a type of public key encryption which is proved to be secure under decisional Diffie–Hellman (DDH) assumption [18]. It has been acknowledged that ElGamal is homomorphic to multiplication, whereas Zether leverages the additive

TABLE 1: Features in different schemes: a comparative summary.

Schemes	Auction type	Cryptographic tools	Auction platform	Privacy	Scalability
THEMIS [4]	VCG	Paillier	Auctioneer	✓	×
PROST [5]	Double	Paillier + OT + GC	Auctioneer + agent	✓	×
ARMOR [6]	Combinatorial	Paillier + OPE + GC	Auctioneer + agent	✓	×
PS-TAHES [7]	Double	Paillier + OT + GC	Auctioneer + agent	✓	×
SLISA [13]	Double	Secret sharing + GC	Auctioneer + agent	✓	×
CREAM [8]	Single	N/A	Ethereum smart contract	×	×
Wang et al. [9]	Single	Paillier + SGX + PC	Ethereum smart contract	✓	×
SAFE [11]	Single	SGX + ZKCP	Ethereum smart contract	✓	×
Ours	Combinatorial	ElGamal + Σ -Bullets	Account-based BC + ZSC	✓	✓

Here, “OT” stands for oblivious transfer. “OPE” is order-preserving encryption. “PC” means Pedersen commitment, and the “ZKCP” represents zero-knowledge contingent payment [14]. Account-based BC can be any blockchain platform operating under the account model, such as Ethereum, Fabric, and so on.

homomorphic feature of ElGamal, so it can be used to hide the balance in exponent.

Let b and b' be two amounts that need to be protected and y be the public key. The ciphertexts can be computed as

$$\begin{aligned} C_L &= g^b y^r, C_R = g^r; \\ C'_L &= g^{b'} y^{r'}, C'_R = g^{r'}. \end{aligned} \quad (1)$$

Then, the encryption of $b + b'$ under y can be calculated as

$$C_L C'_L = g^{b+b'} y^{r+r'}; C_R C'_R = g^{r+r'}. \quad (2)$$

3.4. Σ -Bullet Zero-Knowledge Proof [12]. To ensure the encrypted transactions are correct, Zether provides with a novel ZK proof, namely, Σ -Bullets. Σ -Bullets combine Bulletproofs and Σ -protocols to make algebraically encoded form as $\exists x: g^x = y \wedge h^x = u \in \mathbb{G}$.

A ZK proof for the statement $st: \{(p, q, t, \dots; l, m, n, \dots): f(p, q, t, \dots; l, m, n, \dots)\}$ means that the prover shows knowledge of l, m, n, \dots s.t. $f(p, q, t, \dots; l, m, n, \dots)$ is true, where p, q, t, \dots are public variables.

4. Zether-Based E-Auction Scheme

4.1. System Overview. Figure 1 illustrates a combinatorial spectrum auction via Zether in 5G HetNet scenario. We briefly introduce the workflow of our proposed scheme. In 5G HetNet, a buyer may be interested in more than one spectrum resource. Moreover, different buyers may be interested in a same spectrum resource simultaneously. This is because same spectrum frequency band can be reused in accordance with different geolocations. Therefore, a conflict graph over combinatorial spectrum sets should be constructed firstly. Afterwards, buyers could seal the bids according to their interests in the spectrum combinations. The bids would be locked into the ZSC. ZSC initiates spectrum auction and announces the winner. It should be noted that there may be multiple winners when they have no conflict of interest. At last, winners will be assigned the corresponding spectrum resources and the rest of the bids will be returned to the accounts, respectively. In

addition, we also give some important notations in this paper, as shown in Table 2.

4.2. Detailed Construction. In this paper, we propose a combinatorial spectrum auction in 5G HetNet scenario based on Zether. As shown in Figure 2, it is mainly composed of five parts. The first is the global setting algorithm, which can create the global parameters when it runs once and deploy the ZSC. The next part is the registration of users participating in the auction. The users register a Zether account and deposit a certain amount. The third part is to execute a specific auction, lock the account to the “Secure Auction Execution (called AUC)” smart contract provided in SAFE [11], and then execute the specific auction process in the fourth part, that is, to transfer the control of the Zether account to AUC. The last part is the settlement of the funds and auction items after the auction ends. AUC is complementary to the price difference between the auction winners based on the final auction price. Through transfer, AUC simply burns the entire amount and keeps a part of it (the winner’s payment) and refunds the balance of the remaining bidders.

4.3. Setup. The setup algorithm refers to Setup_1 and Setup_2 subalgorithms. These two subalgorithms are the setting algorithms of the proof mechanism and the signature scheme, respectively. The setting of the proof mechanism may depend on the relationship of the construction of the proof, which means that its correctness would be publicly verified. During the specific implementation, Bulletproofs [19] and Schnorr signatures [20] are utilized, both of which have untrusted settings.

The formal description of the setup algorithm is shown in Figure 3. In addition to deploy the proof and signature mechanism, it also initializes the account table $f(\text{acc})$ as well as the pending transfer list $p(\text{Transfers})$. The last transfer period table lastRollOver is to record recent account updates, the lock table is to record the address when the account is locked, the counter table ctr is to prevent replay attacks, and the variable btotal is to record the total funds of ZTH contracts controlled by the account. Besides, the setup designs an epoch length E and a maximum funds MAX .

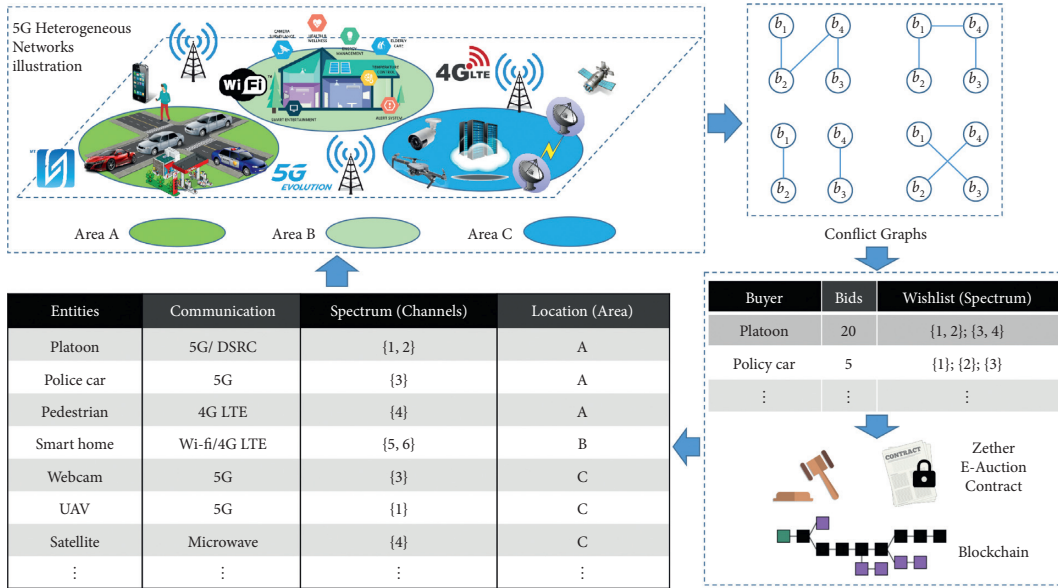


FIGURE 1: System overview.

TABLE 2: Key notations.

Notations	Descriptions
GroupGen	A polynomial-time algorithm where input 1^λ outputs (p, g, \mathbb{G})
p	$p = \Theta(\lambda)$, p is prime
g	A generator of \mathbb{G}
\mathbb{G}	A group of order p
\mathbb{Z}_p	Integers modulo p
y	Public key
σ_{lock}	Signature
acc	Account tables
pTransfers	Pending transfers table
E	An epoch length
Max	A maximum amount value
(ω, p)	Successful auction combination
$(C_{L,i}, C_{R,i})$	Encrypted amounts linked to key y_i

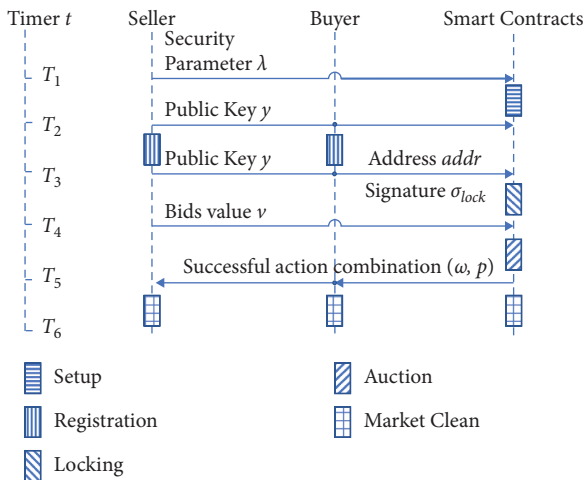


FIGURE 2: Workflow of e-auction via ZSC.

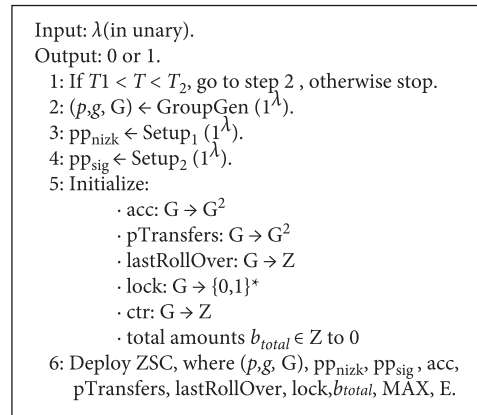


FIGURE 3: Setup.

4.4. Registration. Anyone can fund the account by straightforward specifying the public key y as well as depositing certain ETH. The transfer algorithm is introduced in Figure 4. Transfer transfers ZTH from one account to another, and π_{transfer} ensures that the ciphertext is in the correct form and the transferor has enough money. In addition, there is a signature to avoid replay attacks. As shown in Figure 5, Fund converts ETH to ZTH. ETH is stored in SC, and ZTH is also added to the (pending) balance of y . If the account does not already exist, a new account will be created.

4.5. Locking. Every transaction made to an account is linked with add. The Lock algorithm is introduced in Figure 6. If the account is unlocked, you can do it from any address. However, if you lock to an address, you can only operate from add. CheckLock is an internal method to check both states. Before operating the account, all methods will invoke CheckLock. When it has y , addr, and σ_{lock} , Lock will check whether the account is operated by calling CheckLock.

4.6. Auction Execution. In Figure 7, in the auction execution stage of $T_4 < T < T_5$, the auction execution agreement first checks the deposit amount of each bidder, and the bids of bidders who do not have sufficient deposits will be ignored. Then, the auction execution agreement selects winners and payment amounts for different auction formats. Failed buyers and failed sellers do not have to pay any fees.

In combined auctions, the auction execution protocol sorts \mathcal{V} in descending order $v^1 \geq v^2 \geq \dots \geq v^n$ and greedily distributes the items in order hereafter. Buyer B_i can win her package, if the package does not include any items that have been distributed in the previous winning package. When there are no bidders or available items, the allocation stops. The key bidder is selected as B^c , and its previous bidder is the winner with the smallest bid value, namely, $v^{c-1} \in \mathcal{W}, v_{i,j}^{c-1} \leq v_{i',j'}, \forall B_{i'} \in \mathcal{W} \wedge a^j, a^{j'} \in \mathcal{A}$.

4.7. Market Cleaning. During the market cleaning, bidders and sellers run market clearing agreements to exchange their cryptocurrencies and auction items. After that, the smart contract updates the deposit record of the bidder. Since no bidder has suspended the auction, the SC refunds all funds based on records.

The way to return the deposit is Burn. As shown in Figure 8, Burn transforms ZTH to ETH, and it verifies the proof π_{burn} and st_{burn} to guarantee that the sender holds correct private key and asks for correct amount. Besides, it checks the signature on the transaction data and the counter value to avoid replay attacks. The most important point is that every transfer and destruction transaction in the auction includes ZK proof to ensure that the transferred or redeemed amount is valid without revealing its true value.

```

Input:  $y, \bar{y}, (C, D), (\bar{C}, \bar{D}) \pi_{\text{Transfer}}, \sigma_{\text{transfer}}$ 
Output: 0 or 1.
1: If  $T_2 < T < T_3$ , go to step 2, otherwise stop.
2: RollOver ( $y$ ).
3: RollOver ( $\bar{y}$ ).
4: Let  $(C_L, C_R) = \text{acc}[y]$ 
5: Require:
    · CheckLock ( $y, \text{msg.sender}$ ) = 1
6: Let  $\text{acc}[y] = \text{acc}[y] (C^{-1}, D^{-1})$ .
7: Let  $p\text{Transfers}[\bar{y}] = p\text{Transfers}[\bar{y}] \circ (\bar{C}, \bar{D})$ .
8: Let  $\text{ctr}[y] = \text{ctr}[y] + 1$ .

```

FIGURE 4: Transfer.

```

Input:  $y$ .
Output: 0 or 1.
1: If  $T_2 < T < T_3$ , go to step 2, otherwise stop.
2: RollOver ( $y$ ).
3: Set  $b = \text{msg.value}$ .
4: Require:
    ·  $-b + b_{\text{total}} \leq \text{MAX}$ 
    · CheckLock ( $y, \text{msg.sender}$ ) = 1
5: If  $\text{acc}[y] = \perp$ :
    · Let  $H = \text{block.number}, e = [H/E]$ 
    · Set  $\text{acc}[y] = (1, 1)$ 
    · Set  $p\text{Transfers}[y] = (g^b, 1)$ 
    · Set  $\text{lock}[y] = \perp$ 
    · Set  $\text{lastRollOver}[y] = e$ 
    · Set  $\text{ctr}[y] = 0$ 
    Else:
    · Set  $p\text{Transfers}[y] = p\text{Transfers}[y] \circ (g^b, 1)$ 
6: Let  $b_{\text{total}} = b_{\text{total}} + b$ .

```

FIGURE 5: Fund.

```

Input:  $y, \text{addr}, \sigma_{\text{lock}}$ 
Output: 0 or 1.
1: If  $T_3 < T < T_4$ , go to step 2, otherwise stop.
2: RollOver ( $y$ ).
3: Require:
    · CheckLock ( $y, \text{msg.sender}$ ) = 1
    · Verifynizk ( $y, (\text{addr}, \text{ctr}[y]), \sigma_{\text{lock}}$ ) = 1
4: Let  $\text{lock}[y] = \text{addr}$ 
    · Let  $H = \text{block.number}, e = [H/E]$ .
5: Let  $\text{ctr}[y] = \text{ctr}[y] + 1$ .

```

FIGURE 6: Locking.

5. Theoretical Analysis

5.1. Transfer ZK Proof. In addition to hiding the transfer amount, anonymous transfers also hide the information of sender as well as receiver in the transfer. When someone transfers money b^* from Ethereum address y to \bar{y} , and he or she wants to hide the both address in a bigger range of public keys, where $y = \{y_1, \dots, y_n\}$, let $(C_{L,i}, C_{R,i})$ be the encrypted amounts linked to key y_i , for $i \in [n]$, then the user creates n ciphertexts $(C_1, D_1), \dots, (C_n, D_n)$ as well as proves that (i) one (j th) encrypts b^* , and another one (ℓ th) encrypts $-b^*$,

```

1: If  $T_4 < T < T_5$ , go to step 2, otherwise stop.
2: For buyer  $B_i$ :
   . If  $\text{DPST}[B_i] \leq \max(V)$  (resp.  $\text{DPST}[G_i] \leq \max(S)$ ), continue the loop, otherwise stop.
   . Lets  $v_i \leftarrow 0$  (resp.  $s_i \leftarrow \infty$ )
3: Updates  $V, S, (W, P)$ 
4: Publishes  $(W, P)$ .

```

FIGURE 7: Auction.

```

Input:  $y, b, \pi_{\text{burn}}$ 
Output: 0 or 1.
1: If  $T_5 < T < T_6$ , go to step 2, otherwise stop.
2: RollOver ( $y$ ).
3: Let  $(C_L, C_R) = \text{acc}[y]$ .
4: require:
   .CheckLock ( $y, \text{msg.sender}$ ) = 1
   .CheckLock ( $y, \text{msg.sender}$ ) = 1
5: Let  $\text{ctr}[y] = \text{ctr}[y] + 1$ .
6: Let  $b_{\text{total}} = b_{\text{total}} - b$ .
7: Do  $\text{msg.sender.transfer}(b)$ .

```

FIGURE 8: Burn.

and the remaining users encrypt 0; (ii) b^* is positive; (iii) the remaining funds in y_j (b^j) are positive too.

We can let $D_1 = \dots = D_n = D$ and use randomness in order to effectively process statement without disclosing j, b^*, ℓ , and b^j . Besides, we introduce split-new variables s_1, \dots, s_n and t_1, \dots, t_n . Value 1 for an s_i indicates that funds are being transferred from y_i and value 1 for a t_j indicates that funds are being transferred to y_j . The user will let these variables be confidential and use them to prove different claims. One of s_1, \dots, s_n and one of t_1, \dots, t_n should be 1. This can prove that any of these variables is 1 or 0, $\sum_i s_i = 1$ and $\sum_i t_i = 1$. Besides, the user proves

$$\prod_{i=1}^n C_i^{s_i} = g^{b^*} \prod_{i=1}^n y_i^{r \cdot s_i}, \quad (3)$$

$$\prod_{i=1}^n C_i^{s_i+t_i} = \prod_{i=1}^n y_i^{r \cdot (s_i+t_i)}, \quad (4)$$

$$C_i^{(1-s_i) \cdot (1-t_i)} = y_i^{(1-s_i) \cdot (1-t_i) \cdot r} \text{ for } i \in [n], \quad (5)$$

$$\prod_{i=1}^n \left(\frac{C_{L,i}}{C_i} \right)^{s_i} = g^{b^j} \left(\frac{\prod_{i=1}^n C_{R,i}^{s_i}}{D} \right)^{\text{sk}}, \quad (6)$$

$$g_{\text{epoch}}^{\text{sk}} = u. \quad (7)$$

s_1, \dots, s_n is 1, and the remaining are 0. Equation (3) indicates that the ciphertext for s_i is an effective encryption for b^* . As equation (3) is subtracted from equation (4), $\prod_{i=1}^n C_i^{t_i} = g^{-b^*} \prod_{i=1}^n y_i^{r \cdot t_i}$, which indicates that the ciphertexts of t_i is an effective encryption for $-b^*$. Thereby, both

equations (3) and (4) indicate that the ciphertext-encoded quantities are effective.

In equation (5), $(1-s_i)(1-t_i)$ is non-zero in the case when both s_i and t_i are 0. As the equation shows, i and C_i are an encryption of 0. Equation (6) denotes b^j amounts of the account for which s_i is 1. Finally, equation (7) denotes that u is the surefire random number during the current epoch.

In addition, users need to prove $g^{\text{sk}} = \prod y_i^{s_i}$, $b^*, b^j \in [0, \text{MAX}]$, and the equation associates the secret key with the public key (latter is not revealed), while the latter two equations denote that the transferred amount and the remaining amount are in the correct range. To summarize, users prove the following statement: $\text{st}_{\text{AnonTransfer}}: (y_i, C_{L,i}, C_{R,i}, C_i)_{i=1}^n, D, u, g, g_{\text{epoch}}; \text{sk}, b^*, b^j, r, (s_i, t_i)_{i=1}^n$:

$$\prod_{i=1}^n C_i^{s_i} = g^{b^*} \prod_{i=1}^n y_i^{r \cdot s_i}, \quad (8)$$

$$\prod_{i=1}^n C_i^{s_i+t_i} = \prod_{i=1}^n y_i^{r \cdot (s_i+t_i)}, \quad (9)$$

$$D = g^r, \quad (10)$$

$$\left(C^{(1-s_i) \cdot (1-t_i)} = y_i^{(1-s_i) \cdot (1-t_i) \cdot r} \right)_{i=1}^n, \quad (11)$$

$$\prod_{i=1}^n \left(\frac{C_{L,i}}{C_i} \right)^{s_i} = g^{b^j} \left(\frac{\prod_{i=1}^n C_{R,i}^{s_i}}{D} \right)^{\text{sk}}, \quad (12)$$

$$g^{\text{sk}} = \prod_{i=1}^n y_i^{s_i}, \quad (13)$$

$$g_{\text{epoch}}^{\text{sk}} = u, (s_i \in \{0, 1\}, t_i \in \{0, 1\})_{i=1}^n, \quad (14)$$

$$\sum_{i=1}^n s_i = 1, \sum_{i=1}^n t_i = 1, b^* \in [0, \text{MAX}], b^j \in [0, \text{MAX}]. \quad (15)$$

Finally, $\text{st}_{\text{AnonTransfer}}$ is expressed as equations (8) to (15). The statement is very complicated, but the structure is actually very deep. It turns out that the size can be logarithmic in the range and anonymity set. This is completed by integrating multiple proofs with Bulletproof to encrypt 0.

5.2. Correctness. The algorithms `CreateTransferTx` and `CreateBurnTx` scroll all public keys y according to the status of the SC. Therefore, any unfinished transfers linked with these keys will be returned to the corresponding account, and these unfinished lock requests will become effective. Then, generate the transactions of transfer and burn for the new status of the account, which matches the status of ZSC used to handle them.

Trusted users only place accounts locked to the same address in their anonymous set. When the account holders change the lock of their accounts through calling both methods of lock or unlock, both methods will set the new

TABLE 3: Comparison of different auction schemes in gas consumption among twenty bidders.

Stages	Consumption in gas		
	SAFE [11]	CREAM [8]	Ours
Registration	455395	2357366	262286
Allocation	0	396742	1279664
Clearing	1955742	91442	750200
Total	2411137	2845550	2292150

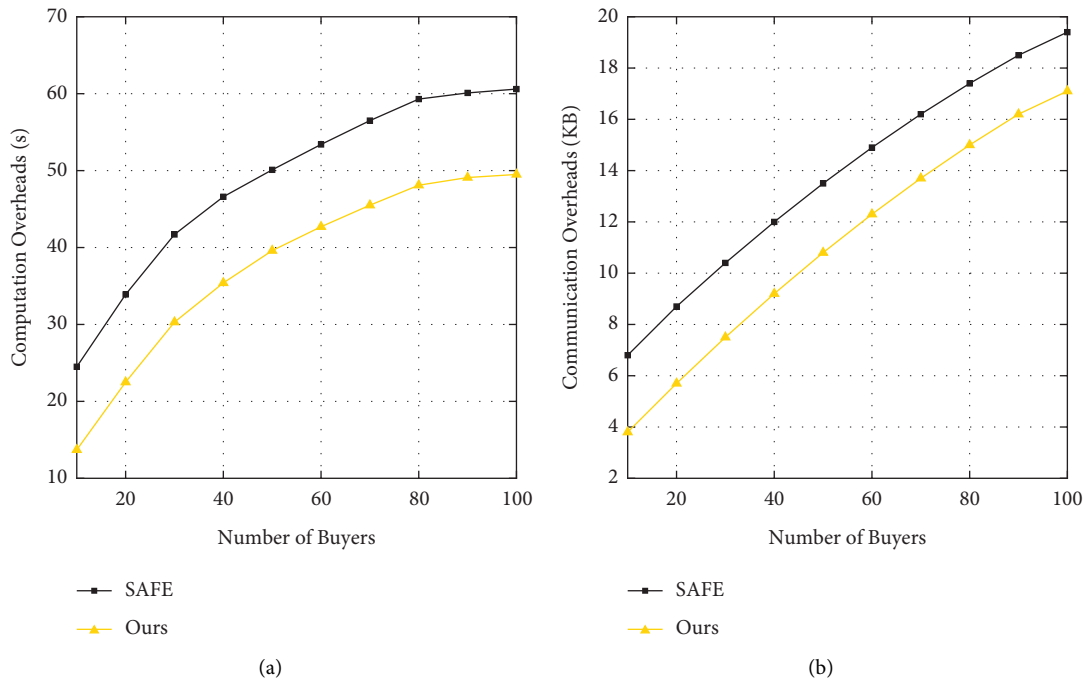


FIGURE 9: The impact of the quantity of buyers' overheads on (a) computation and (b) communication.

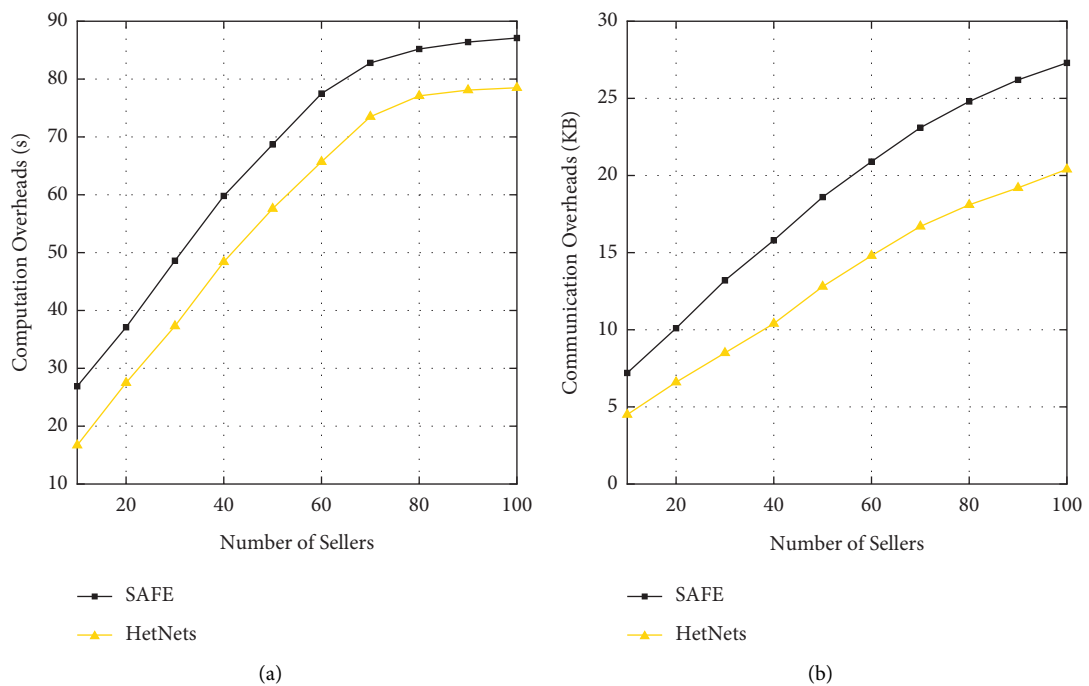


FIGURE 10: The impact of the quantity of sellers' overheads on (a) computation and (b) communication.

lock address as a suspended lock. Therefore, transactions generated during this period will not have an impact.

The remaining of the correctness is the ElGamal encryption in homomorphic properties and the proof system. Although the encrypted value is in \mathbb{Z}_p and ideally deals with positive integers, it has no effect because ZSC only accepts deposits with the maximum amount of MAX, and the constant is smaller than p . Therefore, homomorphic operations will not lead to overflow.

5.3. Experiment Performance. In order to correctly evaluate the proposed smart contracts and show their feasibility, we run contracts in the form of SC. This implementation indicates that our scheme is practical and it is able to run on the Ethereum Virtual Machine (EVM). In order to show the superiority of our scheme, we also compared it with existing research work.

Our proposed implementation of the Ethereum-based smart contract is written in the Solidity language, and some observations have been analyzed and utilized. Ethereum recently introduced a precompiled contract for elliptic curve operations on the BN-128 curve. Compared with direct implementation, these precompiled contracts lower the cost of performing these operations. The reason is that miners can utilize special software to execute these functions more efficiently. These operations are initially introduced to support pair-based ZK-SNARK. Σ -Bullets do not need to be paired. Curve BN-128 is not the best choice for the efficiency or safety of Bulletproofs Σ -Bullets. Despite this, we still choose to use this curve to implement experiment because it is natively supported and the implementation cost is relatively lower.

5.4. Gas Consumption and Overheads. We first measured the gas consumption used to implement basic contract operations. We measure gas consumption including registration, allocation, and clearing. As shown in Table 3, in a combined auction with 20 bidders, combined with SAFE [11] and CREAM [8], our scheme consumed the least gas.

As shown in Figures 9 and 10, we evaluated the system overhead of quantity of buyers and sellers, As buyers' quantity increases but is less than the quantity of items in the combined auction, time and storage consumption will soar because of the high growth rate of the winners. When the quantity of buyers' items exceeds the quantity of buyers, the quantity of winners will remain unchanged, resulting in an increase in overhead during the market clearing phase. The increase in the quantity of sellers or projects will increase the quantity of winners, and the main expense is in the market clearing phase. Overall, our solution has better performance than SAFE.

6. Conclusions

In this paper, we present a Zether-based approach in dealing with the combinatorial spectrum e-auction challenges in 5G HetNet. The e-auction is executed in the ZSC without adopting SGX. Besides, our approach also achieves bidding

value preservation without introducing time-consuming cryptographic tools such as Paillier homomorphic encryption, garbled circuits, and so on. We deploy our approach on Ethereum and testify the effectiveness as well as scalability. Given that gas consumptions in some auction phases are higher than that in state-of-the-art research, we leave these in our future work.

Data Availability

The experimental data required for this article cannot be shared at this time because these data are also part of ongoing research.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this article.

Acknowledgments

This research was supported by MF2009 Project (Trusted Joint Computing on Cross-Border Data) and MOST-FDCT Projects (0058/2019/AMJ) (Research and Application of Cooperative Multi-Agent Platform for Zhuhai-Macao Manufacturing Service).

References

- [1] G. Forecast, "Cisco visual networking index: global mobile data traffic forecast update, 2017–2022," *Update*, vol. 2017, p. 2022, 2019.
- [2] P. Wang, B. Di, H. Zhang, K. Bian, and L. Song, "Cellular V2X communications in unlicensed spectrum: harmonious coexistence with VANET in 5g systems," *IEEE Transactions on Wireless Communications*, vol. 17, no. 8, pp. 5212–5224, 2018.
- [3] A. Networks, "Field test report: Aviat participates in field testing on 6 ghz unlicensed devices with ameren and epri," 2021, <https://blog.aviatnetworks.com/field-test-report-aviat-participates-in-field-testing-on-6-ghz-unlicensed-devices-with-ameren-and-epri/>.
- [4] M. Miao Pan, J. Jinyuan Sun, and Y. Yuguang Fang, "Purging the back-room dealing: secure spectrum auction leveraging paillier cryptosystem," *IEEE Journal on Selected Areas in Communications*, vol. 29, no. 4, pp. 866–876, 2011.
- [5] Q. Wang, J. Huang, Y. Chen, C. Wang, F. Xiao, and X. Luo, "\$PROST\$: privacy-preserving and truthful online double auction for spectrum allocation," *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 2, pp. 374–386, 2019.
- [6] Y. Chen, X. Tian, Q. Wang, M. Li, M. Du, and Q. Li, "ARMOR: a secure combinatorial auction for heterogeneous spectrum," *IEEE Transactions on Mobile Computing*, vol. 18, no. 10, pp. 2270–2284, 2019.
- [7] Q. Wang, J. Huang, Y. Chen, X. Tian, and Q. Zhang, "Privacy-preserving and truthful double auction for heterogeneous spectrum," *IEEE/ACM Transactions on Networking*, vol. 27, no. 2, pp. 848–861, 2019.
- [8] S. Wu, Y. Chen, Q. Wang, M. Li, C. Wang, and X. Luo, "Cream: a smart contract enabled collusion-resistant e-auction," *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 7, pp. 1687–1701, 2019.

- [9] J. Wang, N. Lu, Q. Cheng, L. Zhou, and W. Shi, "A secure spectrum auction scheme without the trusted party based on the smart contract," *Digital Communications and Networks*, vol. 7, no. 2, pp. 223–234, 2020.
- [10] Z. Zhou, X. Chen, Y. Zhang, and S. Mumtaz, "Blockchain-empowered secure spectrum sharing for 5g heterogeneous networks," *IEEE Network*, vol. 34, no. 1, pp. 24–31, 2020.
- [11] Y. Chen, X. Tian, Q. Wang, J. Jiang, M. Li, and Q. Zhang, "Safe: a general secure and fair auction framework for wireless markets with privacy preservation," *IEEE Transactions on Dependable and Secure Computing*, vol. 2020, Article ID 3045449, 2020.
- [12] B. Bünz, S. Agrawal, M. Zamani, and D. Boneh, "Zether: towards privacy in a smart contract world," in *Proceedings of the International Conference on Financial Cryptography and Data Security*, pp. 423–443, Springer, Berlin, Germany, February 2020.
- [13] K. Cheng, L. Wang, Y. Shen, Y. Liu, Y. Wang, and L. Zheng, "A lightweight auction framework for spectrum allocation with strong security guarantees," in *Proceedings of the IEEE INFOCOM 2020-IEEE Conference on Computer Communications*, pp. 1708–1717, IEEE, Toronto, Canada, July 2020.
- [14] M. Campanelli, R. Gennaro, S. Goldfeder, and L. Nizzardo, "Zero-knowledge contingent payments revisited: attacks and payments for services," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, pp. 229–243, New York, NY, USA, October 2017.
- [15] M. B. H. Weiss, K. Werbach, D. C. Sicker, and C. E. C. Bastidas, "On the application of blockchains to spectrum management," *IEEE Transactions on Cognitive Communications and Networking*, vol. 5, no. 2, pp. 193–205, 2019.
- [16] D. Tygar, "Auction types," 2021, <https://www.usenix.org/legacy/publications/library/proceedings/ec98/fullpapers/harkavy/harkavyhtml/node2.html>.
- [17] Wikipedia, "Combinatorial auction," 2021, <https://en.wikipedia.org/wiki/Combinatorialauction>.
- [18] Y. Tsiounis and M. Yung, "On the security of elgamal based encryption," in *Proceedings of the International Workshop on Public Key Cryptography*, pp. 117–134, Springer, Berlin, Germany, May 1998.
- [19] B. Bünz, J. Bootle, D. Boneh, A. Poelstra, P. Wuille, and G. Maxwell, "Bulletproofs: short proofs for confidential transactions and more," in *Proceedings of the 2018 IEEE Symposium on Security and Privacy (SP)*, pp. 315–334, IEEE, San Francisco, CA, USA, May 2018.
- [20] C.-P. Schnorr, "Efficient identification and signatures for smart cards," in *Proceedings of the Conference on the Theory and Application of Cryptology*, pp. 239–252, Springer, Houthalen, Belgium, April 1989.

Research Article

Dominant Feature Selection and Machine Learning-Based Hybrid Approach to Analyze Android Ransomware

Tanya Gera ¹, Jaiteg Singh ¹, Abolfazl Mehbodniya ², Julian L. Webber ³,
Mohammad Shabaz ^{4,5} and Deepak Thakur ¹

¹Chitkara University Institute of Engineering and Technology, Chitkara University, Chandigarh, Punjab, India

²Department of Electronics and Communication Engineering, Kuwait College of Science and Technology, Kuwait, Kuwait

³Graduate School of Engineering Science, Osaka University, Toyonaka, Osaka, Japan

⁴Arba Minch University, Arba Minch, Ethiopia

⁵Department of Computer Science Engineering, Chandigarh University, Punjab, India

Correspondence should be addressed to Tanya Gera; tanya.gera@chitkara.edu.in, Jaiteg Singh; jaiteg.singh@chitkara.edu.in, and Mohammad Shabaz; mohammad.shabaz@amu.edu.et

Received 26 August 2021; Revised 20 September 2021; Accepted 1 October 2021; Published 9 November 2021

Academic Editor: Jie Cui

Copyright © 2021 Tanya Gera et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Ransomware is a special malware designed to extort money in return for unlocking the device and personal data files. Smartphone users store their personal as well as official data on these devices. Ransomware attackers found it bewitching for their financial benefits. The financial losses due to ransomware attacks are increasing rapidly. Recent studies witness that out of 87% reported cyber-attacks, 41% are due to ransomware attacks. The inability of application-signature-based solutions to detect unknown malware has inspired many researchers to build automated classification models using machine learning algorithms. Advanced malware is capable of delaying malicious actions on sensing the emulated environment and hence posing a challenge to dynamic monitoring of applications also. Existing hybrid approaches utilize a variety of features combination for detection and analysis. The rapidly changing nature and distribution strategies are possible reasons behind the deteriorated performance of primitive ransomware detection techniques. The limitations of existing studies include ambiguity in selecting the features set. Increasing the feature set may lead to freedom of adept attackers against learning algorithms. In this work, we intend to propose a hybrid approach to identify and mitigate Android ransomware. This study employs a novel dominant feature selection algorithm to extract the dominant feature set. The experimental results show that our proposed model can differentiate between clean and ransomware with improved precision. Our proposed hybrid solution confirms an accuracy of 99.85% with zero false positives while considering 60 prominent features. Further, it also justifies the feature selection algorithm used. The comparison of the proposed method with the existing frameworks indicates its better performance.

1. Introduction

Ransomware has blown away the cyber security world in recent past. It targets the major losses like data, money, and even life. These are special malware used to extort money in return of access and data without user's consent. Attackers are consistently working on producing advanced methods to deceit the victim and generate revenue. According to coalition's cyber insurance claim report (Cyber Insurance Claims Report, 2020), out of 87% reported attacks, 41% are due to ransomware attacks as shown in Figure 1. The possible reason for this significant increase is because of

COVID-19 pandemic; most of the employees are working remotely. The rapidly changing nature and distribution strategies along with smart tactics are also responsible for deteriorated performance of primitive ransomware detection techniques. Ransomware is generally seen in two forms: locker-ransomware and crypto-ransomware [1]. Locker-ransomware attacks lock the victim's device to restrict its use until they pay ransom. On the other side, crypto-ransomware attacks encrypt all personal files to make them inaccessible for owner. Victims are forced to pay ransom to allow unrestricted access to their own personal and confidential data. To classify, analyze, and detect malicious application

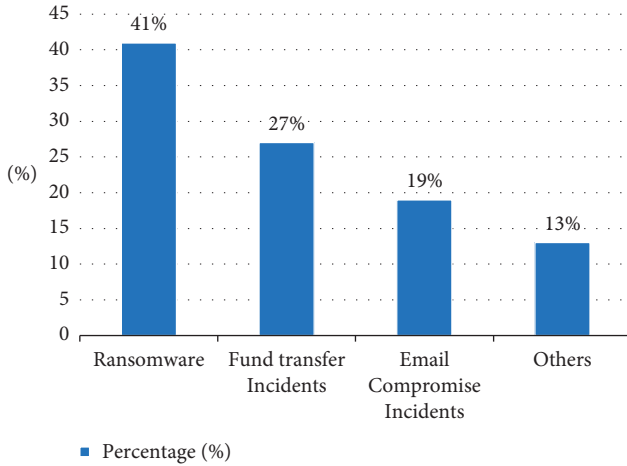


FIGURE 1: Ransomware share in recent reported cyber incidents (modified from Cyber Insurance Claims Report, 2020).

samples, there exists use of two primitive approaches, i.e., static and dynamic techniques. Static techniques examine the applications by matching their signature, code, or permissions used and can detect previously known ransomware only.

Though the literature witnessed that static analysis is fast and effective in detection of Android ransomware, static analysis techniques are popular for their ability of identifying only known ransomware. Considering fast-evolving nature of Android ransomware, static analysis is not enough. Static analysis is based on code and signature similarity and fails at code obfuscation. On the other hand, dynamic analysis checks the general behavior of an application while execution. Dynamic analysis techniques are strong enough to withstand with vulnerable situations and can even detect suspicious behavior even when code is compressed or encrypted. However, dynamic analysis also has a few flaws against smart malware tactics being used these days. Smart malware actions are sometimes triggered only under certain conditions, which is not possible to achieve in emulated testing environment. Hence, fusion of effective static techniques with dynamic techniques could give a robust hybrid solution for Android ransomware. The literature also states that there exist comparatively less studies based on hybrid technique for identifying Android ransomware. Existing hybrid solutions majorly vary in feature set used for detection of Android ransomware. Most of the hybrid approaches focus on a specific ransomware family or a specific ransomware type or specific feature only. Those type-specific or family-specific solutions would be difficult to consider as a generalized solution. Another important aspect to be considered here is novelty required in collecting and utilizing the important features for analysis. Ransomware families utilized new evolving features for constructing new variants, hence creating need of constructing robust feature set for analysis and detection. The success of any approach directly depends on feature set and feature selection method being used. To accurately classify the applications, the feature set being used has to be well-built. Extracting prominent

features and feature selection methods to be used is an ongoing research challenge. Suspicious authors constantly modify a few features to make frequent new variants, hence posing challenge for existing techniques. However, most of the existing studies focus on one or two types of features only for their analysis and detection while testing its run-time behavior. Though system calls, permissions and APIs are important features to be used for analysis and detection of Android ransomware. However, the literature lacks in kernel level checks, file operations, system component, phone state, and so on. Researcher often faces difficulty in predicting all possible behavior set due to limited availability of ransomware dataset and its fast-evolving nature.

In this work, we performed static analysis as well as dynamic analysis over the collected sample of 3249 clean and malicious applications. Static analysis was performed using Apk tool. In the static feature extraction phase, we focus on manifest file to extract permissions associated with the application sample. In parallel, dynamic analysis was performed over the collected data samples using an emulator, i.e., habo analysis system. During the dynamic feature extraction phase, we focussed on API calls, system calls, permissions, file operations, network features, and other system components. Further, static and dynamic feature vectors were transformed to build combined feature matrix containing unique features. A novel feature selection algorithm was applied to select k -prominent features iteratively. Multiple machine learning classifiers were applied to classify samples as clean or ransomware.

The major contributions of this work are follows:

- (i) This work demonstrated the effective use of obtained dynamic features by studying combined impact of all the dynamic features. To the best of our knowledge, prior existing studies utilized one or two standard features like system calls and API. Here, we focussed on all the significant obtained dynamic features to build the efficient dynamic model.
- (ii) We have also built a dominant feature selection algorithm to extract top k -dominant features being used by Android ransomware samples and clean sample. This helped to discriminate among risky and nonrisky features to effectively analyze malicious behavior.
- (iii) With exhaustive experimentation by varying the number of features to be 20, 40, 60, and till 80, we showed the absolute difference in nominal frequency of features used by Android ransomware and clean applications.
- (iv) We evaluated the effectiveness of machine learning classifiers by calculating accuracy, false positive, and false negative rate of each classifier for different set of features iteratively. The result shows that among all the machine learning algorithms, random forest algorithms achieved the highest accuracy of about 99.85% with zero false negative.
- (v) We have also compared the results of our proposed method with those of the existing system as shown in Table 1. The results of our proposed hybrid

TABLE 1: Comparison with existing studies.

Reference	Approach	Machine learning model used	Feature set used	Accuracy (%)
[2]	Static	Random forest, logistic regression, XGBoost, Naive Bayes, support vector machine (SVM), deep learning, and decision tree classifier	Intent, permission, API calls, system commands, and malicious activities	96.3
[3]	Dynamic	Naive Bayes, SVM, and logistic regression	Application programming interface (API)	97
[4]	Hybrid	SVM	Permission, API calls, system calls	99.7
	Our proposed method (dynamic)	J48, LMT, random forest, and random tree	System calls, system components, system command, phone events, run-time permissions, and broadcast receivers	99.85

framework outperform the existing static, dynamic, and hybrid approaches. Our proposed hybrid solution confirms the accuracy of 99.85% with zero false positives while considering 60 prominent features. Further, it also justifies the feature selection algorithm used.

The rest of the paper is organized as follows. The second section presents related work. The complete methodology followed is explained in the third section. The fourth section presents experimental results followed by conclusion and future scope in the last section.

2. Related Work

Two prominent approaches to restrict ransomware infections are static and dynamic analysis of software applications. Static analysis investigates the structural properties of an application without executing it. It primarily emphasizes on code, metadata, and digital signatures imbued within software [5–7]. On the contrary, dynamic analysis examines application behavior by executing it. It executes software within a simulated environment and studies its behavior. Application behavior corresponds to the access permissions, network usage, and information shared, processed, and exchanged by the software application during execution. Static analysis requires less resources and is fast. However, they got failed in case of code obfuscation. On the other hand, dynamic approaches are more effective in performing actual behavior check. However, dynamic approaches are incapable of executing all possible paths and also cannot check interapplication communication on emulators. Hence, many researchers have also worked on hybrid approaches to increase the performance of ransomware detection.

Reference [8] attained lot of popularity and success because in their methodology, they make use of multiple properties together for the analysis and detection. They used source code as well as permissions for capturing static features. This model achieved better performance results by exploring feature level granularity through API calls. Reference [9] proposed that a significant static approach developed was based on application features for detection of malware. It captures important permissions and suspicious API calls of applications, assigns a weight value to them, and then compares it with a threshold value

so as to make appropriate decisions. Weight value for each application is based on the nature of the identified malicious patterns. Reference [10] gave a signature-based static technique. Its aim was to scan the payload to check the threatening strings relevant to financial claim. However, this technique was not much popular because generally text messages for financial claim are sent from C&C (Command and Control) server. Reference [2] proposed framework consists of multiple layers for filtrations. In this paper, they generate a message digest value, i.e., MD5 (message digest) based on suspicious permission being used, dangerous permissions being granted, and hazardous intentions. Appropriate decisions are further made on basis of hash value comparison. Reference [3] focussed on checking whether any file had undergone any remarkable changes. Authors make use of techniques like content similarity and entropy measurement for performing the checks. Reference [4] framed a static model capable of identifying both locker as well as crypto-ransomware. This model does not require any apk to be decompiled because its detection is based on bytecode of application. It does not make use of source code. It also can detect the multiple variants of ransomware. Reference [11] built a static model called R-PackDroid that was light weight solution and was implemented on users' device itself. Its functionality was to extract and analyze the application packages from the apk files. Reference [12] extended their previous work, which has attained a considerable improvement. For the successful implementation of their designed experiment, they gained the administrative rights by rooting the device. After getting the root access, they performed extensive testing on the several applications like financial applications and social applications. Their experiments observed that most of the crucial applications do not fulfil the minimum-security requirements, which increase the chance of data leakage. Reference [13] made use of hierarchical steps of analysis before installation of an application to guarantee its trustworthiness. It has the capability of labelling each application in one of categories as either trustable or type of risk associated with it, i.e., high risk, low risk, and medium risk. For its successful implementation, its analysis is based on multiple information being gathered like permissions used by applications, number of downloads, source of the application, and its rating and developer reputation also. This

approach does not include code-based detection as it used only application metadata. They proved their approach as an effective as well as reasonable approach. Reference [14] developed a completely automatic malware identification mechanism. Its results are based on the multiple classifiers which categorize each application as benign or malicious with the appreciable accuracy of 82.93%. For their experimental observations, they used a very large set of applications containing 107,327 safe and 8,701 malicious applications along with the feature set of top 34,630 out of 23,74,340 features. To maintain balance between the performance and results of all the classifiers, they collaborate performance of all. Reference [15] used supplementary techniques that have always played important role when combined with conventional techniques. Here, in this paper also, authors have firstly captured the metadata of each application and their associated features like developer info, number of downloads, application creation date and time, and permissions being granted. Then, further it applies appropriate machine learning algorithms to assess and analyze. This is a simple and effective approach to gain high performance accuracy. The literature also suggests deep learning feature fusion for identifying mobile malware [16]. Research trends in Android literature have been performed by authors and suggest that machine learning has ability to achieve better accuracy [17, 18]. According to authors [19], healthcare organizations are the key targets of ransomware attack due to the vitality and confidentiality of patient data and then comes the governmental institutions as criminals know the importance of data for the government and they expect to get back the ransom. The third main target of ransomware attack is higher educational institutions due to weak IT hierarchy and then comes the law firms and mobile users who become the target of ransomware attack. Table 2 shows rank-wise targets organizations affected by ransomware attacks.

Reference [20] formulated a hybrid technique called as MONET which is based on the static as well as dynamic analysis. In this model, behavior of the user is consistently monitored and mapped against the run-time behavior of the malicious application. It also includes signature matching generated on the basis of API calls. The significant aim of this approach was to identify malware as well as its variants. Reference [21] attempted to provide full protection against malware, and most importantly this model gives descriptive analysis to users about the threat and its awareness measures. This model sustained high performance accuracy as it is a three-step fold mechanism. It makes use of combined benefits of multiple approaches like static and dynamic and further merged it with effects of machine learning algorithms or local-remote hosts. First, it includes static analysis using a famous framework called Drebin [22] feature set. It also then applied dynamic analysis with the use of system calls which actually improves their analysis results. Further, it applies appropriate machine learning concepts and local-remote host concepts to strengthen their performance accuracy. Reference [23] observed that library component does contain some instances of its malicious behavior. Based on

TABLE 2: Key targets of ransomware attacks.

Rank targets	Key target organisations
1	Healthcare sector
2	Government institutions
3	Education
4	Law firms
5	Mobile and MAC users

this apparent observation, authors developed a unique approach in which they detect the malwares on the basis of abnormal library instances. The major part of the whole process emphasizes on to find whether a library instance has been renamed or not. For the demonstration of their framework, they used more than 1100 applications set out of which 185 were found to be malicious as their library instances were found to be abnormal. Reference [24] proposed a new framework to perform malware detection on the basis of network traffic flow. They considered all the constraints of the traditional static and dynamic techniques such as code obfuscation and resource limitation. In comparison to which, they find that their approach seems to be quiet promising. Based on the fact that most of malware develop and spread across the multiple devices during network processing, so analyzing the network flow will definitely help in identification of malicious activities associated with applications. The proposed approach performs automatic feature selection using appropriate natural language processing and achieves 99.15% detection rate. The framework was also claimed to perform better than many antivirus scanners.

The literature witnesses that the most of the existing frameworks consider system calls, API tracing, and static features like manifest files and permissions, for detection and analysis. On the contrary, ransomware families target the other features and also target personal information and device information. Towards the end of 2017 (Quick heal, 2018), it was reported that ransomware is making use of unique features and make frequent new variants. Examples are doubleLocker that locks both screen as well as data. Some of variants show smart behavior, and their action is based on the Internet status of the user. Such frequently emerging new features which had never been seen before pose a great challenge for existing techniques [1]. However, the engineering new feature fusion method to support in-depth study of all ransomware families is the need of the hour.

3. Methodology

This section presents the overall methodology followed for the hybrid framework to mitigate Android ransomware. We have included details of data collection, feature extraction, feature selection, and machine learning classifiers. To enhance the effectiveness of the proposed hybrid framework, we have built the feature selection algorithm to extract k -dominant features. This proposed hybrid framework also utilizes the various machine learning models to classify each apk file as ransomware or clean.

3.1. Data Collection. In this experimentation, applications are collected from two major sources. For clean applications, around 1486 apk files have been downloaded from Google Play Store. Google Play Store is an official Android market that promises to provide the most trusted source of applications. Google Play Store developer and support team claim that they do not permit applications which mine the cryptocurrencies [25]. For malicious data samples, Android Malware Dataset (AMD) is used. AMD is a standard repository which officially provides access to its dataset especially for research purpose [26] and has been used by many researchers in their study [27–29]. AMD provides updated and latest release for its collection. AMD dataset contains thousands of malicious applications. In this work, we have included only ransomware families which cover 1763 ransomware samples. Here, in this study, Android applications are termed as clean applications or ransomware applications as in Table 3.

3.2. Proposed Hybrid Framework. Figure 2 presents the overall methodology of the proposed hybrid framework. A large set of 3,249 application samples containing both clean and malicious samples are used as input. First, static analysis is performed on each application in data sample to extract static features associated with that application. Further, dynamic analysis is performed to extract dynamic features set used by both benign and ransomware applications. Static analysis and dynamic analysis are performed in parallel to extract feature set. Further, we transformed the obtained static and dynamic feature set information to build a combined feature vector matrix. A well-designed feature selection algorithm is applied to identify k -dominant features. This algorithm is applied iteratively to identify k -dominant feature where k is set to be 20, 40, 60, and 80. To evaluate the effectiveness of model, machine learning models are applied to train and classify each apk files as clean or ransomware application.

3.3. Feature Extraction. In this work, we majorly focus on manifest.xml file to extract static properties associated with that application. Manifest file provides metadata like package name, acquired permissions, and related application components, i.e., activities, broadcast receivers, and other services required as static properties only hold features being used without executing an app. For advanced cyber-attacks, it becomes important to check actual behavior analysis of application. Hence, in this work, we also performed dynamic analysis of each apk file to extract the dynamic features. To evaluate the effectiveness of static and dynamic techniques over Android ransomware applications, we analyzed a few applications statically as well dynamically. We observed the similarity in feature usage pattern among clean as well as ransomware samples. Common features are considered to be the most dangerous features. It becomes very important to scan those static and dynamic features for better results. This laid the formation of the algorithm for extracting dominant features for our proposed hybrid framework as discussed in subsequent sections.

3.3.1. Static Feature Extraction. For extraction of static properties associated with application, we have used Apk tool version 2.4.0 [30] as shown in Figure 3. Apk tool is a popular open-source tool that decompiles apk file to extract its code and other metadata details. The decompressed files contain manifest.xml, resource folder, and java code. Permissions are generally considered to be one of the most important static properties. Each application acquires a set of permissions upon installation. These permissions can be easily extracted from manifest.xml file. In this work, python scripts are used to extract permissions using Apk tool. Apk tool decompiles each apk file, extracts the permissions associated with it, and helps store the information in a text file format. The scripts involve the following steps:

- (i) The script requires .apk file as input
- (ii) It uses Apk tool v2.4.0 to decode .apk file to xml file, dex files, and other resource folder
- (iii) The script scans manifest.xml file to extract all permissions using “permission” tag
- (iv) Further, this information is stored to text file format
- (v) These steps are repeated for all the .apk files

The working of script for static feature extraction using Apk tool is as shown in Figure 4. The steps are repeated for all applications, i.e., apk files in the collected dataset. As the dataset contains both clean and ransomware applications, static permission is analyzed thoroughly. Further, we observed similarity in feature usage between clean and ransomware applications. It was found that permission used by clean applications is quite similar to permissions being used by ransomware samples. Hence, those permissions are considered to be riskiest permission and must undergo checks for analyzing any application. The details of permission extracted are shown in the result section.

Apk tool takes up the largest proportion and is often used to decompile APKs. Current support tools for static analysis and its percentage of use in other studies [31] are enumerated as shown in Figure 5.

A study over Android detection mechanisms using static features also confirms that around 41% of techniques used permissions as a key parameter for detection and analysis of Android malware [31]. Other features used are API calls, metadata, intents, and so on as depicted in Figure 6.

3.3.2. Dynamic Feature Extraction. Dynamic analysis observes the actual behavior of an application while in execution. It is quite obvious that on-device real time execution of Android application on Android platform will result in high consumption of battery and other device resources. Hence, in this study, dynamic analysis is performed on the virtual emulated environment to examine its dynamic features as shown in Figure 7. The literature states that dynamic features like API calls, permissions, and system calls are frequently used features. In this work, emulator called habo analysis system [32] is used which has capability to scan and extract other set of dynamic features also. Security analyst

TABLE 3: Difference in clean and ransomware applications.

	Clean applications	Ransomware applications
Characteristics	These applications do not contain malicious code in the source code. These are safe for device.	These applications do contain malicious code in the source code. Malware authors, i.e., attackers may inject the code to affect the device users.
Installation	Upon installation of clean applications, it performs its dedicated task and does not harm either the device or user's data.	Ransomware applications encrypt the confidential data and file in system upon installation. These can even lock the device and demand ransom to unlock it.

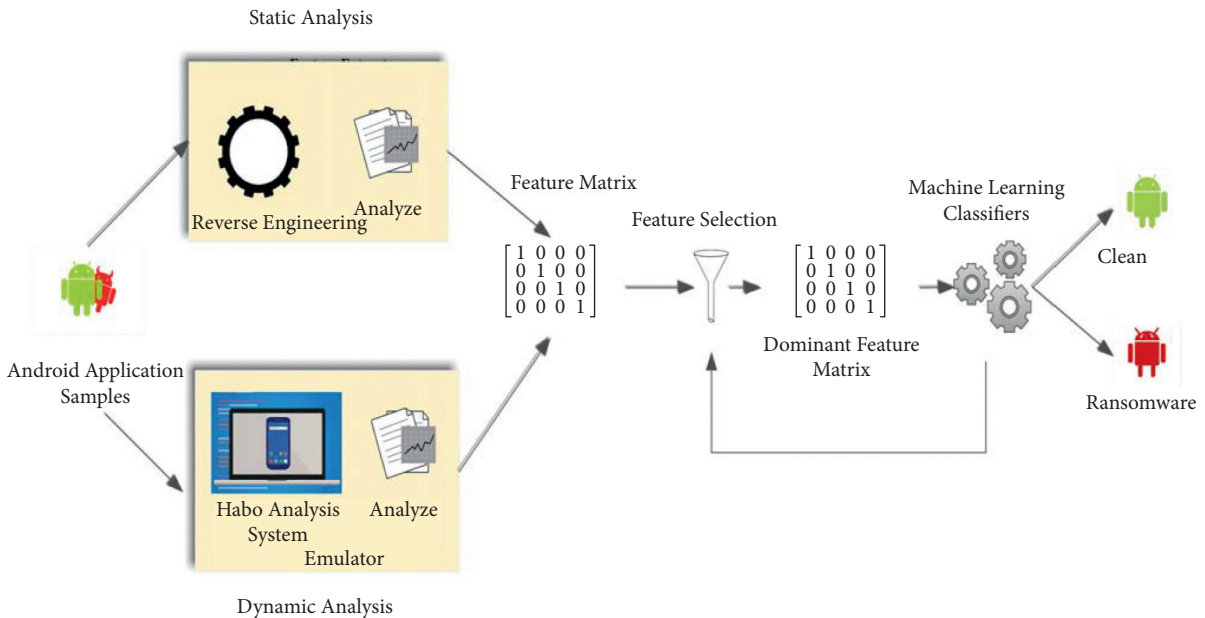


FIGURE 2: Proposed hybrid framework.

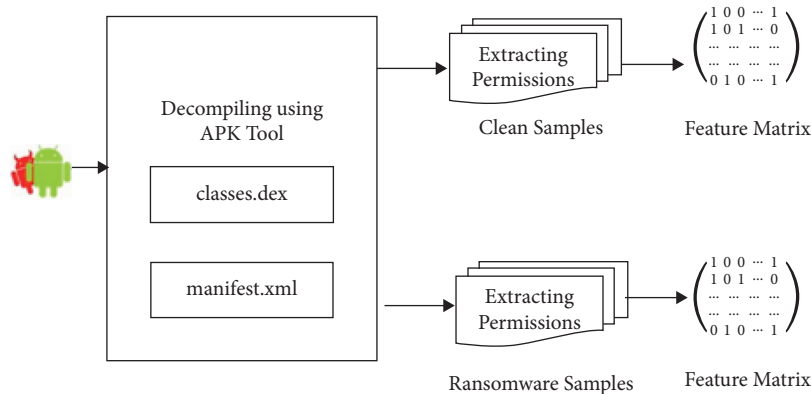


FIGURE 3: Static feature extraction.

generally used the habo analysis system to automate the process of malware analysis. Dynamic features used in this proposed work include API calls, permissions, system calls, network, file monitoring, and other system components. A robust approach to perform an effective dynamic analysis lies in extracting a limited set of features that provide the ability to classify between ransomware and benign behavior of application being tested. For which, we have used the prominent feature selection algorithm as discussed in

subsequent sections. The traces obtained upon execution under controlled virtual environment are recorded to generate the individual reports. These reports contain significant information about dynamic features like API calls, permissions, system calls, network, file monitoring, and other system components. Further, these generated reports are converted to required input format for the experiment. The steps followed for extracting the dynamic features are as follows:

```

I: Using Apktool 2.4.0 on es-file-explorer-4-2-1-9.apk
I: Loading resource table...
I: Decoding AndroidManifest.xml with resources...
I: Loading resource table from file: _WorkArea1\Frameworks\1.apk
I: Regular manifest package...
I: Decoding file resources...
I: Decoding values/ XMLs
I: Baksmaling classes.dex...
I: Baksmaling classes3.dex...
I: Baksmaling classes2.dex...
I: Copying assets and libs...
I: Copying unknown files...
WORKING ON EXTRACTING PERMISSIONS
FINISHED.

```

FIGURE 4: Working of apk tool.

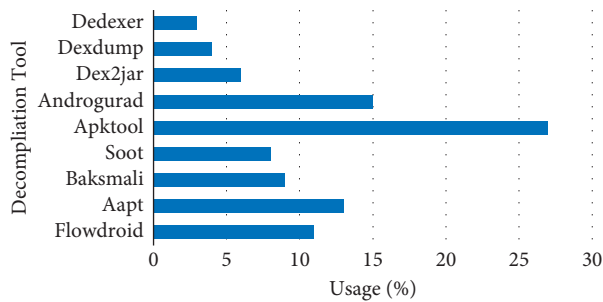


FIGURE 5: Use of apk tool.

- (i) Set up the environment settings
- (ii) Install VirtualBox 5.1
- (iii) Upload the source code to virtual machine to compile it
- (iv) After successful compilation, upload .apk file
- (v) For each application in dataset,
 - (a) Test and analyze the application
 - (b) Download the output.dynamic report file

3.4. Feature Vector. In this step, the recorded features from the previous step were transformed into nominal representation to build feature vector. The feature used by application is marked as 1 denoting its presence and 0 in case of its absence.

Let us assume an application that uses set of features (f_1, f_2, \dots, f_n). For every application in collected dataset, i.e., clean as well as ransomware, f_n is calculated based on formula as follows:

$$f_n = \begin{cases} 1, & \text{if feature exists,} \\ 0, & \text{otherwise.} \end{cases} \quad (1)$$

3.5. Feature Selection Using Prominent Feature Selection Algorithm. Static analysis is based on code and signature similarity and fails at code obfuscation. Dynamic analysis techniques are strong enough to withstand with vulnerable situations and can even detect suspicious behavior even when code is compressed or encrypted. Smart malware actions are sometimes triggered only under certain

conditions, which is not possible to achieve in emulated testing environment. Hence, we have used a fusion of static features with dynamic feature to produce promising results for hybrid solution for Android ransomware. Existing hybrid solutions majorly vary in feature set used for detection of Android ransomware. Ransomware families utilized new evolving features for constructing new variants. The success of any approach directly depends on feature set and feature selection method being used.

The feature set must be unique for both clean feature vector and ransomware feature vector. Hence, a unique feature set is created by taking combination of all the static and dynamic feature sets used by clean samples and ransomware samples. Initially, extracted static features and extracted dynamic features were large in number and redundant. Further, a total of 94 features were extracted as unique set of features as shown in Table 4. Considering all the features or larger set of feature combination for analysis and classification may lead to redundant data. Moreover, to maintain the accuracy and effectiveness of results, we have used prominent feature set in this work. To identify the most significant features, we used a feature selection algorithm as stated Algorithm 1. This algorithm determines top k -dominant features being used by both ransomware and clean applications.

Feature vector files contain data in the form of zeros and ones to represent existence and absence of each feature fed as an input. Further, we calculated sum of frequencies of each feature in clean feature vector file and further normalized it by dividing it with total number of samples, i.e., for clean as well as ransomware samples. The value of nominal frequency for each feature determines its dominance. Then, we calculated the absolute difference between both normalized frequencies for each feature. It represents similarities in feature existence in both clean and ransomware samples. For extracting the most used features, we sorted all the values in ascending order. The smaller values of difference signify more dominance of that feature whereas higher the difference, lesser the dominance of the feature. Initially, we identified the top 20 most dominant features to analyze and classify the samples. However, we have also iteratively increased the number of dominant features by 20 at each step. However, it is expected that considering the large number of feature combination may result in high consumption of system resources as well as time. The difference in nominal frequencies of features among clean and ransomware applications is discussed in the result section.

3.6. Classification Using Machine Learning Models. The obtained combination of unique set of static and dynamic features is used to train machine learning models. In this work, we have used supervised learning. Two class labels used are c for clean application and r for ransomware for training the classification models. Existing solutions [33, 34] have suggested many classifiers and attained promising results. So, during our experiments, we have used multiple classifiers to test and validate our results which includes random forest [35], decision tree (J48) [36], logistic model

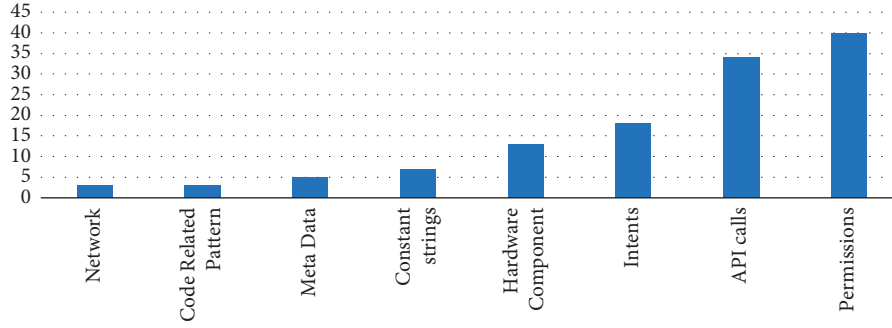


FIGURE 6: Use of permissions.

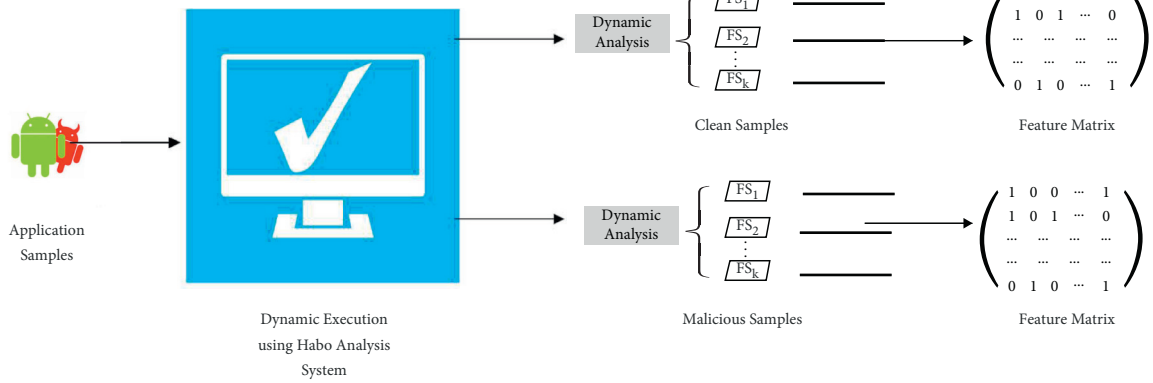


FIGURE 7: Dynamic feature extraction.

tree, and random tree. Selection of the correct number of dominant features was critical decision of the feature selection phase. Initially, top 20 dominant features were selected. Further, experiment was repeated by incrementing dominant features by 20 at each step till 80, i.e., 20, 40, 60, and 80.

3.7. Performance Evaluation. For measuring the performance evaluation statistics, we have used the following metrics.

3.7.1. Accuracy. Accuracy of machine learning models can be found by dividing the total number of correctly classified with sum of actual positives and actual negatives. The formula for calculating the accuracy is as follows:

$$\text{accuracy} = \frac{TP + TN}{TP + TN + FP + FN} * 100. \quad (2)$$

3.7.2. Recall. Recall is fraction of true positive with sum of true positives and false negatives. The equation for calculating recall can be found as follows:

$$\text{recall} = \frac{TP}{TP + FN}. \quad (3)$$

3.7.3. Precision. Precision is division of true positive with sum of true positives and false positives. The equation for calculating precision can be found as follows:

$$\text{precision} = \frac{TP}{TP + FP}. \quad (4)$$

3.7.4. F-Measure. A good score of precision and recall will lead to a good F-measure of the model. This value represents the harmonic mean and justifies the strength of the model for classification. The formula for its calculation is as follows:

$$F - \text{measure} = \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \times 2. \quad (5)$$

4. Experimental Results

The experimental results of this study are discussed in this section. Intense manual analysis over initially obtained feature set helped to identify the most dangerous features used by Android ransomware as discussed in Section 4.2. To determine the relevance and dominance of feature, we analyzed results with varying number of features during the feature selection algorithm as discussed in Section 4.3. Classification results with top k -dominant features and their corresponding performance evaluation are presented in Section 4.4.

TABLE 4: Unique features list extracted.

Feature_No	Feature_Name
f1	Access URL
f2	Access database
f3	Access location
f4	Access mail session
f5	Access network
f6	Access network state
f7	Access shared app data
f8	Activate device manager
f9	Active activity
f10	Active ActivityForResult
f11	Add alert window
f12	Add view
f13	Aquire root access
f14	Call setAction of intent
f15	Change WIFI (wireless fidelity) state
f16	Change component property
f17	Change network state
f18	Check available GPS
f19	Check root access
f20	Create database
f21	Create file
f22	Create new process
f23	Detect device id (antisimulator)
f24	Detect operator brand (antisimulator)
f25	Disable keyguard
f26	Execute SQL query
f27	Execute system command
f28	File read
f29	File remove
f30	Get WIFI state
f31	Get accounts
f32	Get connected WIFI
f33	Get device id
f34	Get installed app
f35	Get last location
f36	Get main intent of apk
f37	Get phone number
f38	Get running service
f39	Get running task
f40	Get scanned WIFI
f41	Get special property of simulator
f42	Get specific account
f43	Get standby state
f44	Get stored WIFI
f45	Get user id
f46	Hide from desktop
f47	Initialize URI
f48	Initialize URL
f49	Initialize intent
f50	Initialize monitor driver file
f51	Initialize new process
f52	Install shortcut
f53	Intercept broadcast
f54	Kill background processes
f55	Launch apk via intent
f56	Load class
f57	Load dynamic library
f58	Load website in webview
f59	Make toast
f60	Monitor network data

TABLE 4: Continued.

Feature_No	Feature_Name
f61	Open bluetooth
f62	Parse URI
f63	Read URL data
f64	Read call log
f65	Read external storage
f66	Read history bookmarks
f67	Read one line from buffer
f68	Read system settings
f69	Receive network data
f70	Record audio or media
f71	Register receiver
f72	Reset password
f73	Run-time error
f74	Scan WIFI
f75	Send broadcast
f76	Send extra information
f77	Send mail via intent
f78	Send network data
f79	Send notification
f80	Send SMS
f81	Set looped task
f82	Set timed task
f83	Start recording
f84	Start service
f85	Stop recording
f86	Uninstall shortcut
f87	Vibrate
f88	Window information
f89	Write external storage
f90	Write file
f91	Write system settings
f92	SetSharedPreferences
f93	AddAppToShareData
f94	ReadSharedPreferences

4.1. Experimental System Setup. Being a hybrid approach, we required a good device and other computational resources for our experiments. It includes both static and dynamic analyses of a large dataset of 3249 application samples. Table 5 shows the details of system setup and tools used during the experiment.

4.2. Feature Extraction and Critical Analysis over Obtained Feature Set. All the static and dynamic execution reports were transformed to feature vector format to analyze obtained features for both clean and ransomware samples. Nominal values for each feature show whether a particular feature is used by that sample or not. Based on reports of clean and ransomware feature vector statistics, we identified top 30 features used by clean samples as well as top 30 features used by ransomware samples as shown in Figures 8 and 9 separately. The results showed that ransomware application sample uses many crucial features also, and moreover a few clean application samples are also used. It becomes cumbersome for analyst to make decisions. For example, our results show that the use of feature Access Network (f5) is 75% by clean applications whereas 82% use

Input: Unique feature vector data for both clean and ransomware samples

Output: List of k -dominant features

Symbols Used: Let S_c be the total number of clean sample, S_m be the total number of ransomware samples, and K be the number of dominant features required to be extracted

Step 1: for all clean samples, calculate sum of frequencies of each feature and normalize it $\text{Normalized_Frequency}_{\text{Clean}}(fi) = \sum_{i=0}^n \text{Frequency}(fi)/S_c$

Step 2: for all ransomware samples, calculate sum of frequencies of each feature and normalize it $\text{Normalized_Frequency}_{\text{ransomware}}(fi) = \sum_{i=0}^n \text{Frequency}(fi)/S_m$

Step 3: for all features in unique feature list, calculate the absolute difference between normalized frequencies of clean and ransomware sample $\text{Diff}_{\text{Normalized_Frequency}(fi)} = \text{Normalized_Frequency}_{\text{Clean}}(fi) - \text{Normalized_Frequency}_{\text{ransomware}}(fi)$

Step 4: Sort $\text{Diff}_{\text{Normalized_Frequency}(fi)}$

Step 5: Choose k to record k number of dominant features for k in (20, 40, 60, 80) iteratively.

ALGORITHM 1: Dominant feature selection.

TABLE 5: Experimental system requirements.

Static analysis tool	Apk tool v2.4.0
Dynamic analysis tool	Habo analysis system
Data mining tool	Weka 3.8.3
Operating system	Windows 10
Processor	Intel(R) core (TM) i5-8250U CPU@ 1.80 GHz
RAM	8.0 B

in ransomware applications samples. Access Network indicates establishing communication with Internet which can be very dangerous in case of ransomware application. Similarly, features like Send Network Data (f78), Receive Network Data (f69), and Send Extra Info (f76) have been observed to be 10–15% more in use than a normal clean application sample. Making communication with command and control servers is the major step involved in ransomware working mechanism. So, it justifies that it is important to check such critical features while analyzing application against ransomware attacks. The use of file operations like File Read (f28) and File Remove (f29) do not differ in large, hence should be added to list of risky features. Based on the observation made, we intend to focus on similarity in feature usage pattern among clean as well as ransomware samples. Common features which do not differ in large are considered to be the most dangerous features. It becomes very important to scan those static and dynamic features for better results. To produce effective results, we have used the feature selection algorithm for extracting dominant features for our proposed hybrid framework as discussed in subsequent sections.

4.3. Feature Selection. Initially, we analyzed all the features of all the samples and found that occurrences of usage of some features in clean and ransomware applications differ in large. To record the difference in nominal frequency of each feature among clean and ransomware samples, we tend to find k -dominant features as discussed in Algorithm 1. Further, we implemented the experiment by varying the value k as 20, 40, 60, and 80. The varying k helped to perform cross-analysis about dominant features over all the collected

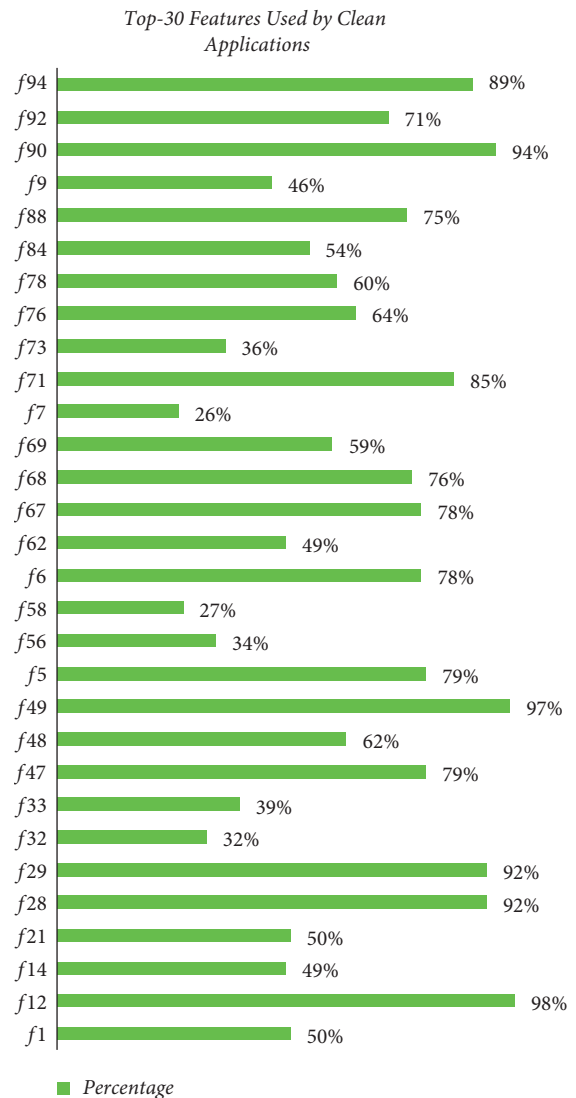


FIGURE 8: Top 30 features used by clean samples.

samples. The dominant features distinguish the differences in the behavior of clean and Android ransomware applications. Here, graphs as in Figures 10–15 represent

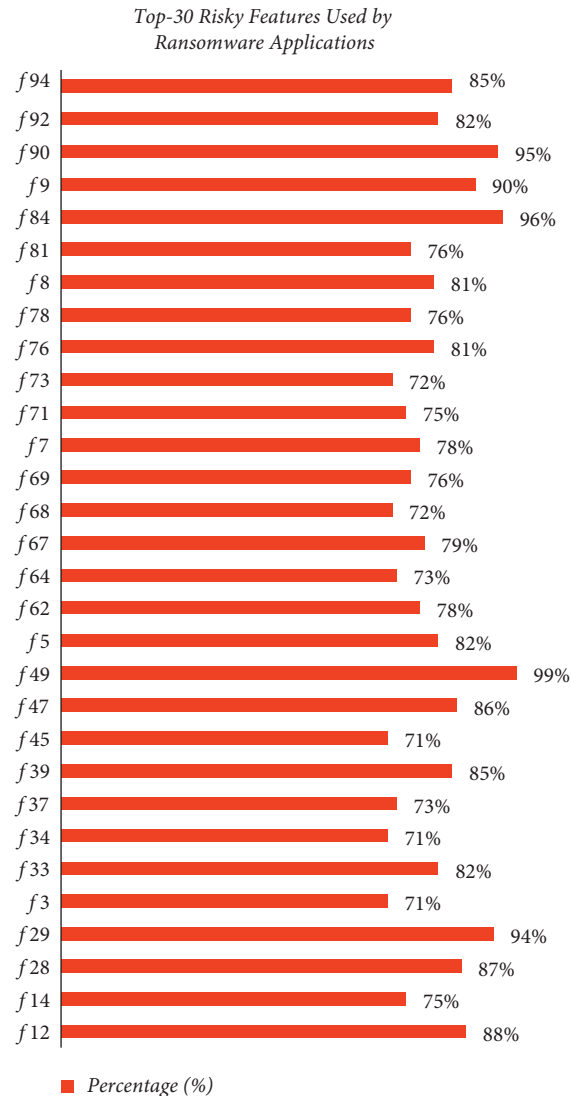


FIGURE 9: Top 30 risky features used by ransomware samples.

difference of normalized feature occurrence for top 20, 40, 60, and 80 dominant features, respectively.

The results of top 20 dominant features extracted include Access URL (f1), Access location (f3), Access shared app data (f7), Activate device manager (f8), Active Activity (f9), Create file (f21), Get device id (f33), Get installed app (f34), Get phone number (f37), Get running service (f39), Get user id (f45), Initialize URL (f48), Load class (f56), Load website in webview (f58), Read call log (f68), Run-time error (f73), looped task (f81), Set timed task (f82), Start service (f84), and Window information (f88). We observed that clean applications generally do not use much of a few features like Activate device manager (f8), Read call log (f68), and Get running service (f39) but ransomware applications do.

However, top 40 dominant features include all the features extracted as top 20 list as well as a few more features like Access Database (f2), Access Network State (f6), Call setAction of intent (f14), and Check root access (f19). Features like Access Database (f2), Access Network State (f6), Call setAction of intent (f14), and Check root access

(f19) are majorly used by ransomware applications to perform kernel level check to attain the root access and device admin privileges. The results also justify that our feature selection algorithm is able to identify the most crucial features which must be included for analysis procedure.

Similarly, we have also identified top 60 and top 80 dominant feature lists for our experiments. Selection of the correct number of dominant features was critical decision of the feature selection phase. Initially, top 20 dominant features were selected. Further, experiment was repeated by incrementing dominant features by 20 at each step till 80, i.e., 20, 40, 60, and 80. To compute the effectiveness of the model, we have applied the classification model iteratively for all top extracted features as discussed in Section 4.4.

4.4. Classification. In this phase, we performed the classification over collected data set containing both ransomware and clean applications. The major purpose is to identify suitable classifier with the appropriate number of features

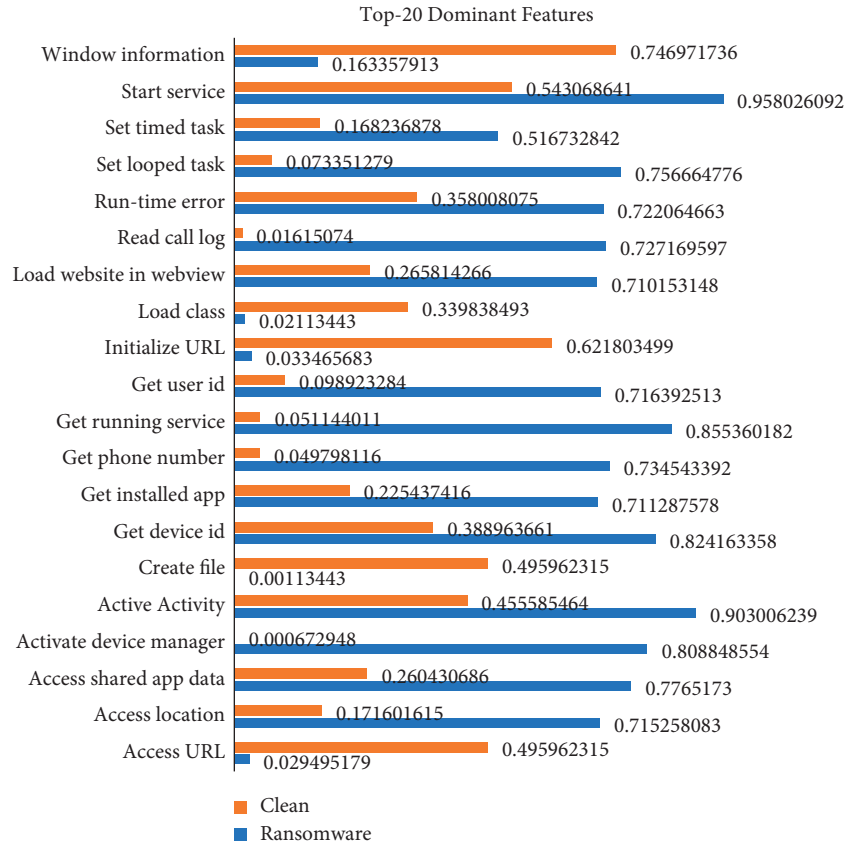


FIGURE 10: Difference in nominal frequencies of top 20 dominant features.

which can classify applications with highest accuracy. We have used multiple classifiers to test and validate our results which include random forest, decision tree (J48), logistic regression, and random tree. During the feature selection algorithm, we decided to extract top k -dominant features with varying value of k to be 20, 40, 60, and 80. Classification results with all values of k are presented in subsequent sections. We evaluated and compared the performance of classifiers with other performance measurement statistics, i.e., accuracy, recall, precision, and F -measure as shown in Figures 16–19.

Figure 20 shows that initially J48 produced highest false positives. Further, with the increase in the number of features set, the considerable dip represents a slight better performance than LMT (logistic model tree) and random tree. Overall, random forest produces minimal values for false positive over the change of the number of features and least when 60 dominant features were considered.

Figure 16 shows that initially random forest, random tree, and LMT produce almost the same values for false negatives. Further, with the increase in the number of features set, the downfall represents a slight better performance. However, J48 produced highest false negatives throughout different sets of dominant features. With k to be 40, random forest produced minimal values for false negatives. The rest gradually becomes stable with varying number of features.

Accuracy of any machine learning classifiers can be calculated by dividing the total number of correctly classified with sum of actual positives and actual negatives. The line chart as shown in Figure 17 illustrates that with the increase in the number of features, there is a substantial increase in performance of all the classifiers. Overall, random forest found to be the best in classifying applications sample into clean or ransomware.

The results show that among multiple classifiers, the random forest algorithm outperforms in terms of highest accuracy, lowest false negative, and false positive for all sets of features taken to be as 20, 40, 60, and 80. With 60 dominant features, random forest algorithms achieved the highest accuracy of about 99.85% with zero false negative. As the random forest algorithm is based on ensemble learning, the problem of overfitting and missing data is reduced. Due to its abundance qualities, it has also been used to detect ransomware by other researchers as the only classifier used in their studies [37, 38]. Researchers also do compare the performance of multiple classifiers, and their results also indicate that random forest performs better than random tree or any other single decision model tree [39].

4.4.1. Classification Results with Top 20 Dominant Features. Figure 18 shows effect of selecting 20 dominant features as an input dataset on F -measure along with the results obtained from computing precision and recall for multiple

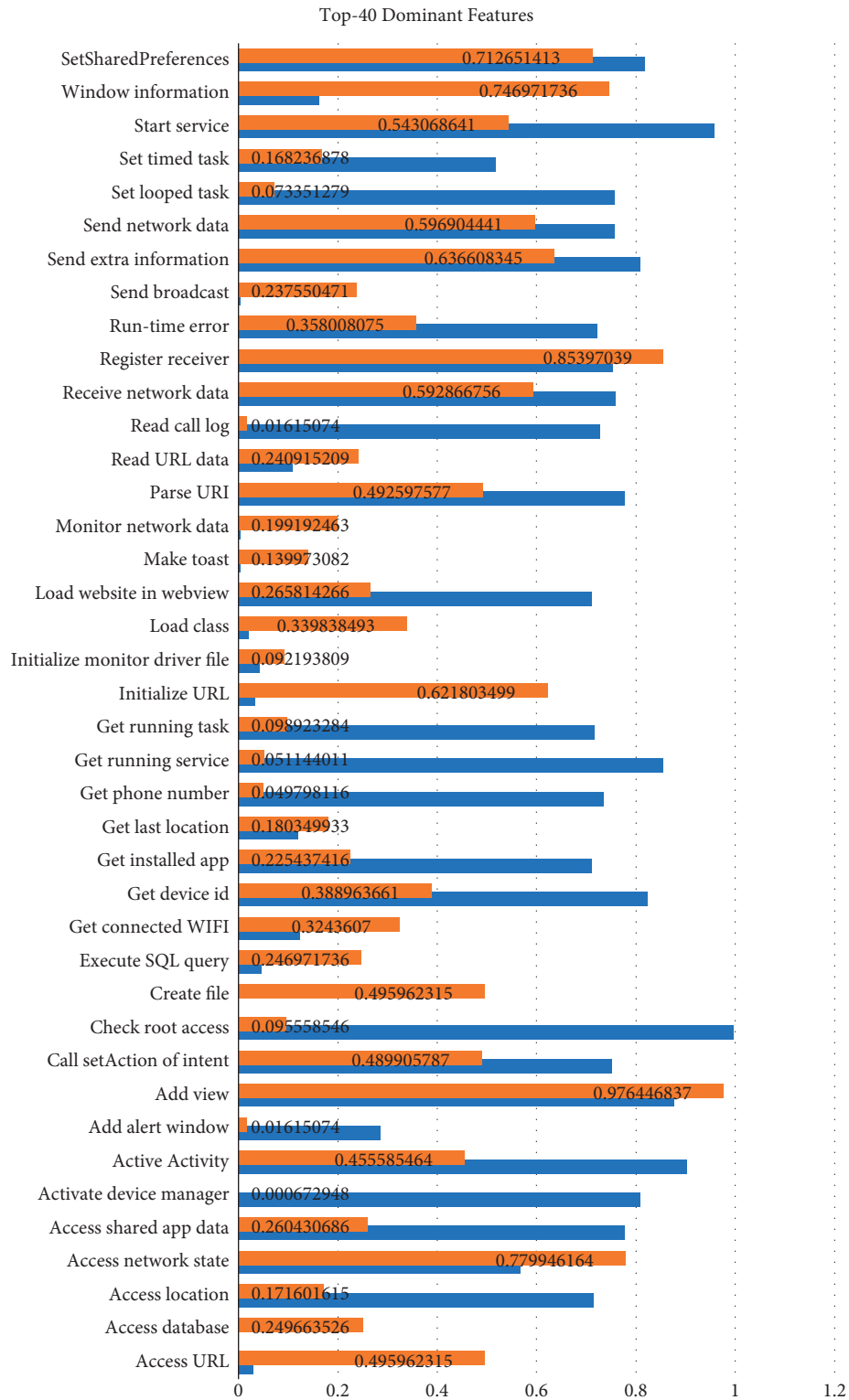


FIGURE 11: Difference in nominal frequencies of top 40 dominant features.

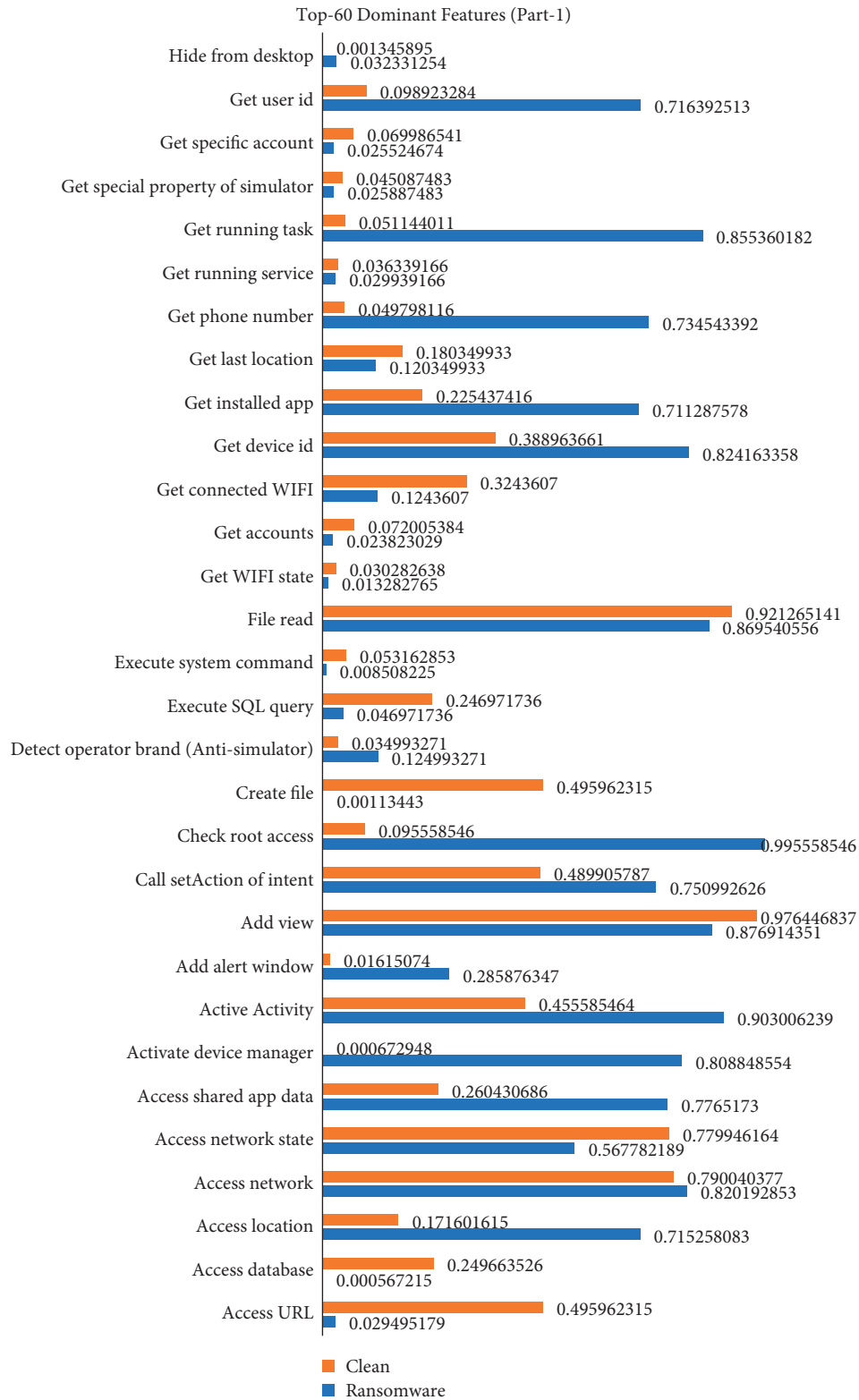


FIGURE 12: Difference in nominal frequencies of top 60 dominant features (Part 1).

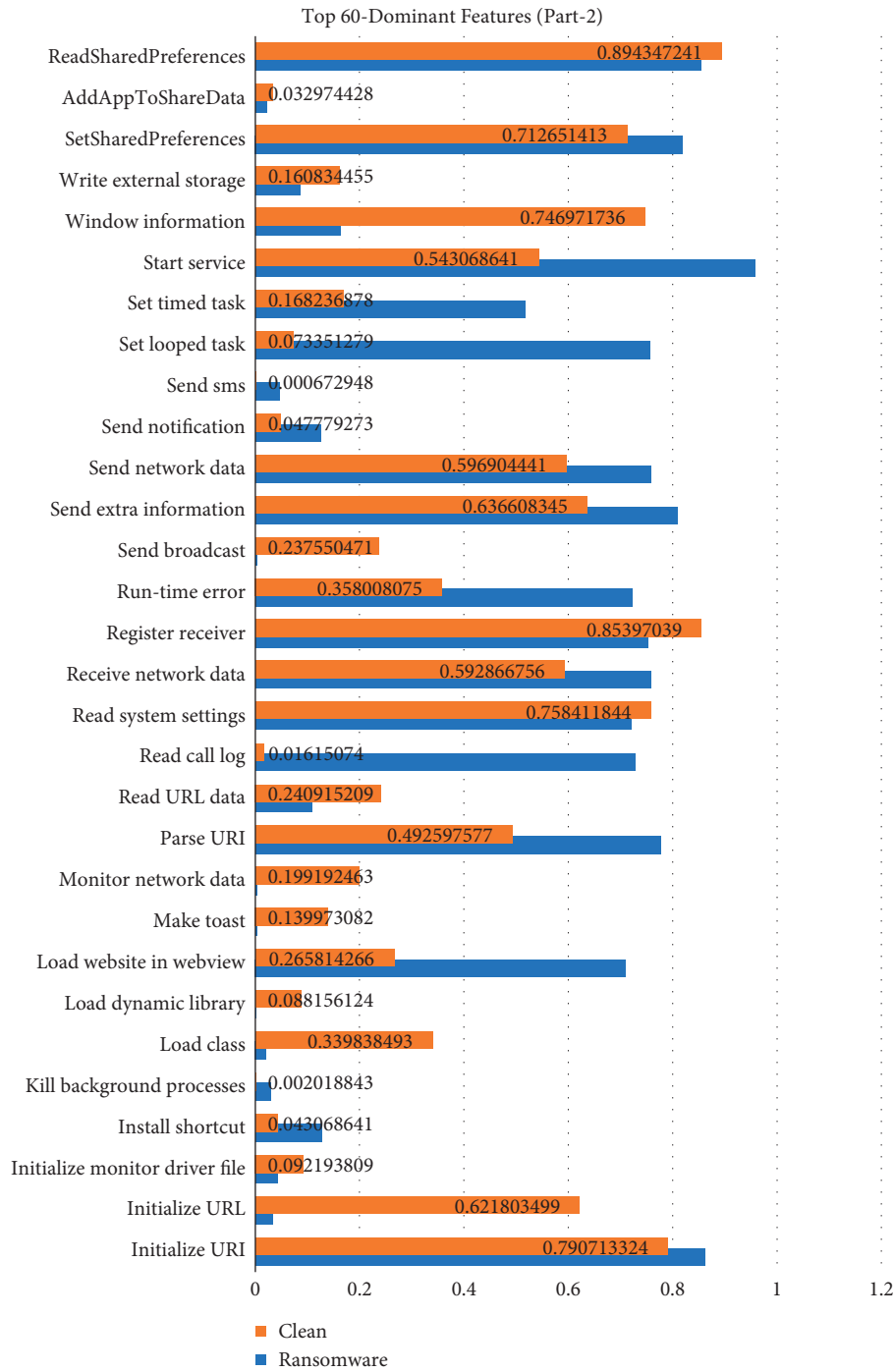


FIGURE 13: Difference in nominal frequencies of top 60 dominant features (Part 2).

classifiers, i.e., J48, random forest, LMT, and random tree. Computational values of random forest, random tree, and LMT are closely equivalent to each other but random forest has achieved best values of recall, precision, and *F*-measure, i.e., 0.984118, 0.986356, and 0.985236, respectively.

4.4.2. Classification Results with Top 40 Dominant Features. Figure 19 shows effect to cater 40 dominant features as an input dataset on *F*-measure, precision, and recall for multiple classifiers, i.e., J48, random forest, LMT, and random tree. Computational values of precision for random forest

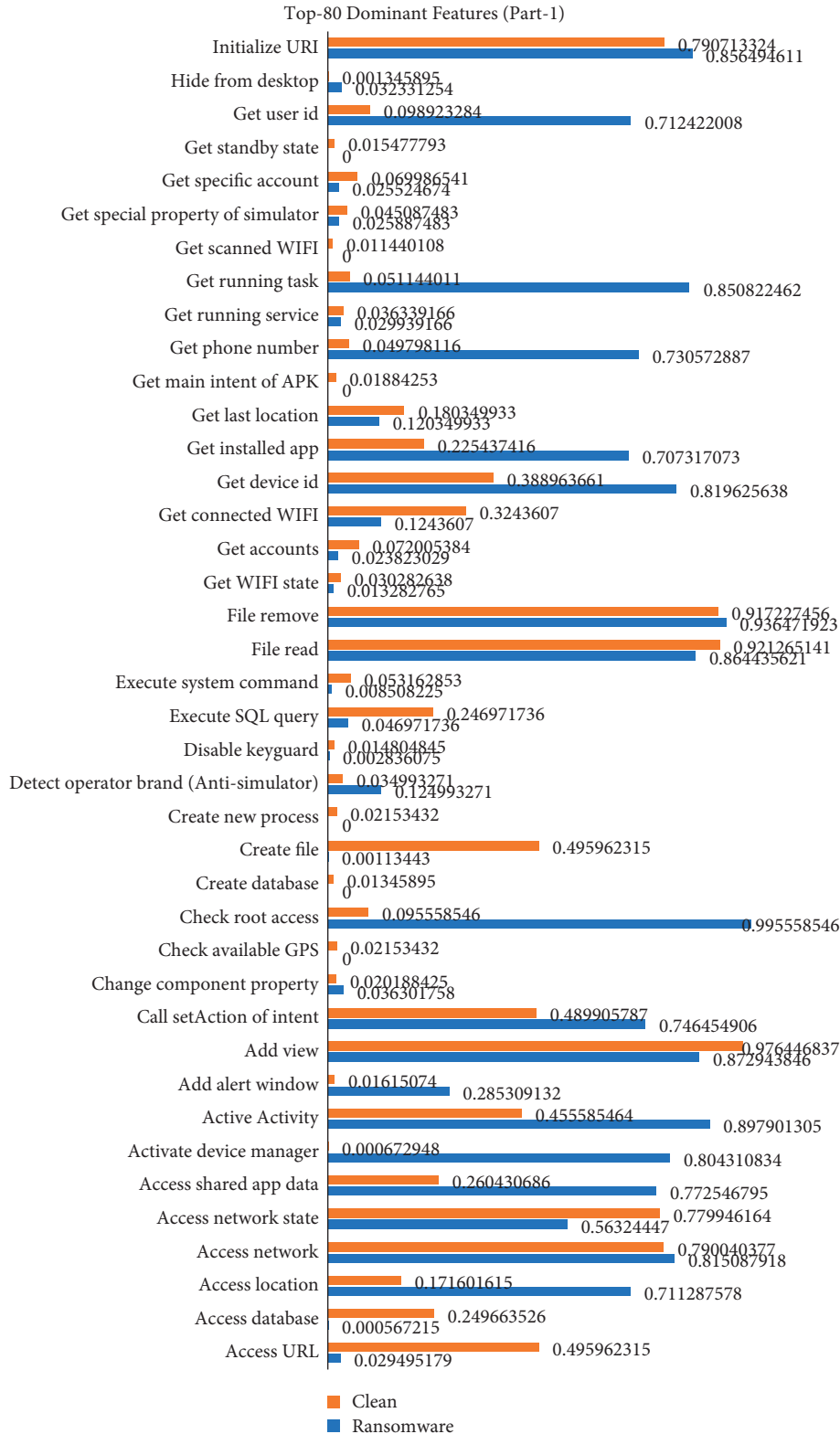


FIGURE 14: Difference in nominal frequencies of top 80 dominant features (Part 1).

and J48 are slightly different to each other but overall random forest has achieved the best values of recall, precision, and *F*-measure, i.e., 0.997731, 0.991545, and 0.994628, respectively.

4.4.3. *Classification Results with Top 60 Dominant Features.* There is dramatic change in results of 60 dominant features. Here, all the classifiers performed significantly well to classify the application samples. Figure 21 shows level up of

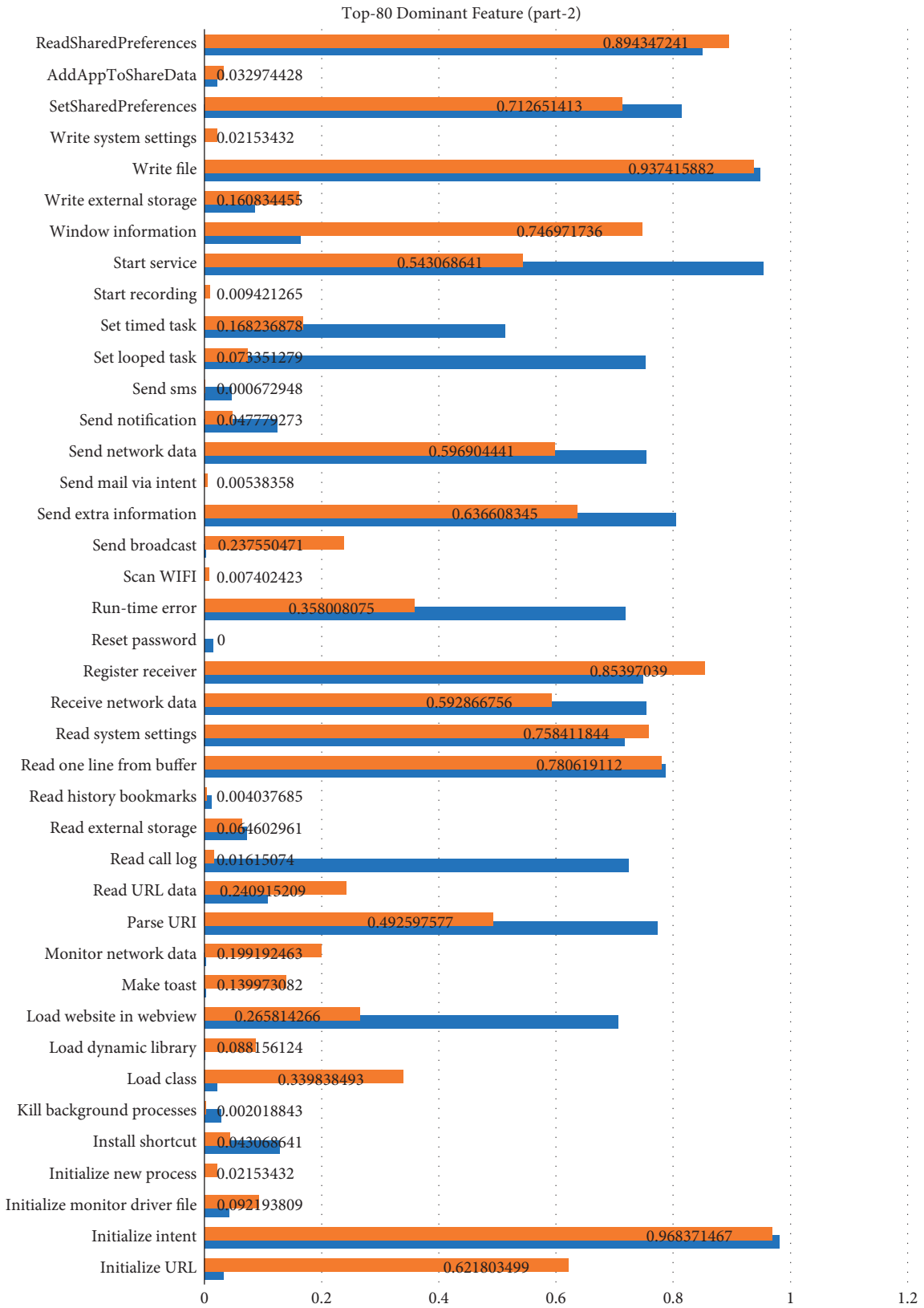


FIGURE 15: Difference in nominal frequencies of top 80 dominant features (Part 2).

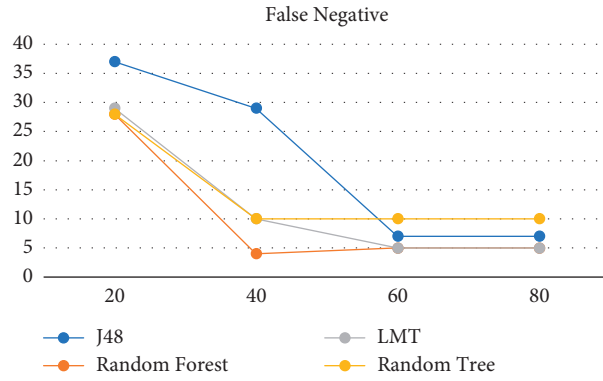


FIGURE 16: False negative rate.

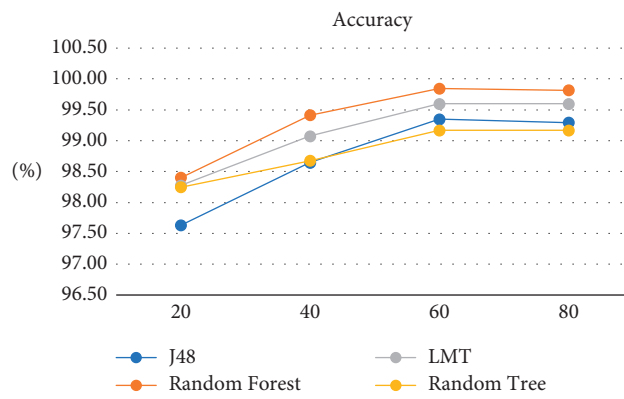


FIGURE 17: Accuracy of multiple classifiers.

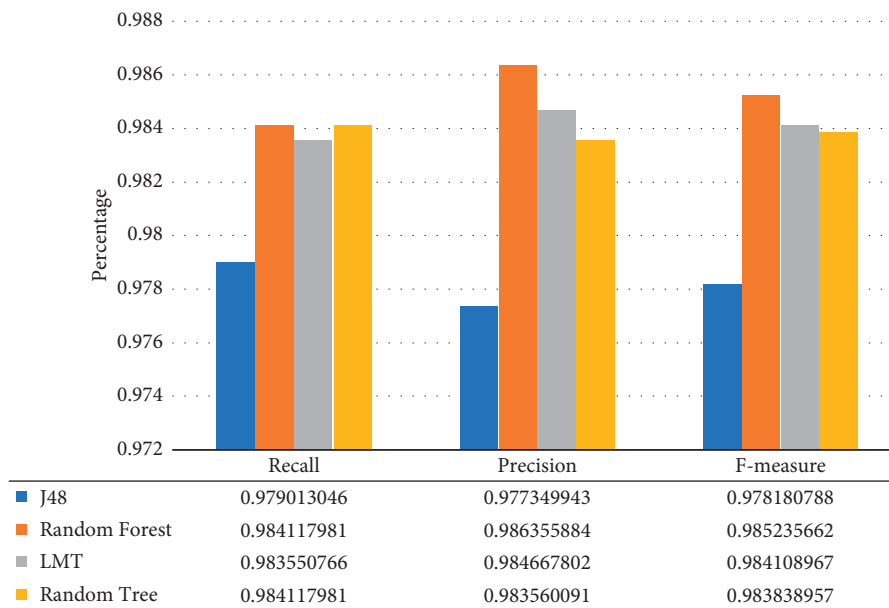


FIGURE 18: Classification results for top 20 features.

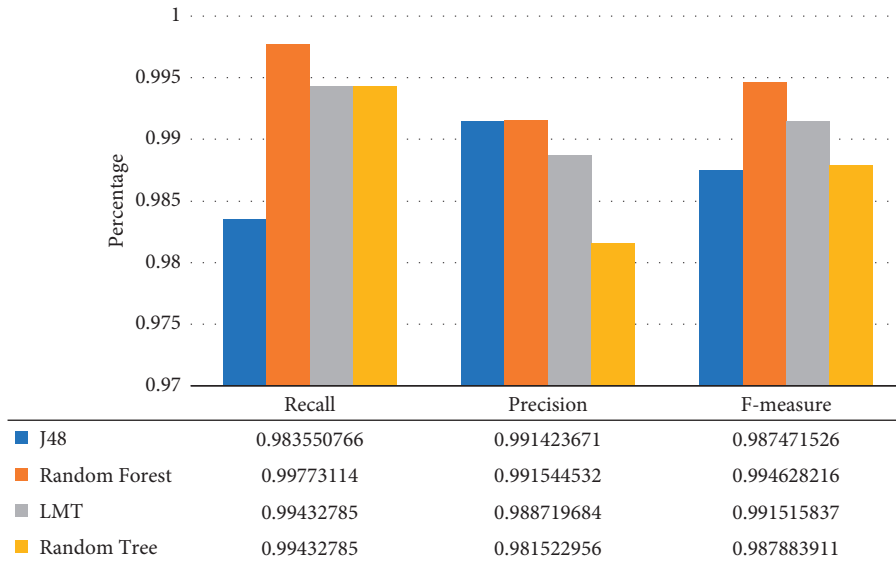


FIGURE 19: Classification results for top 40 features.

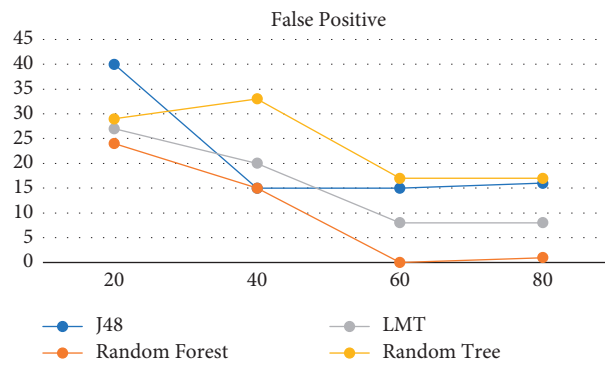


FIGURE 20: False positive rate.

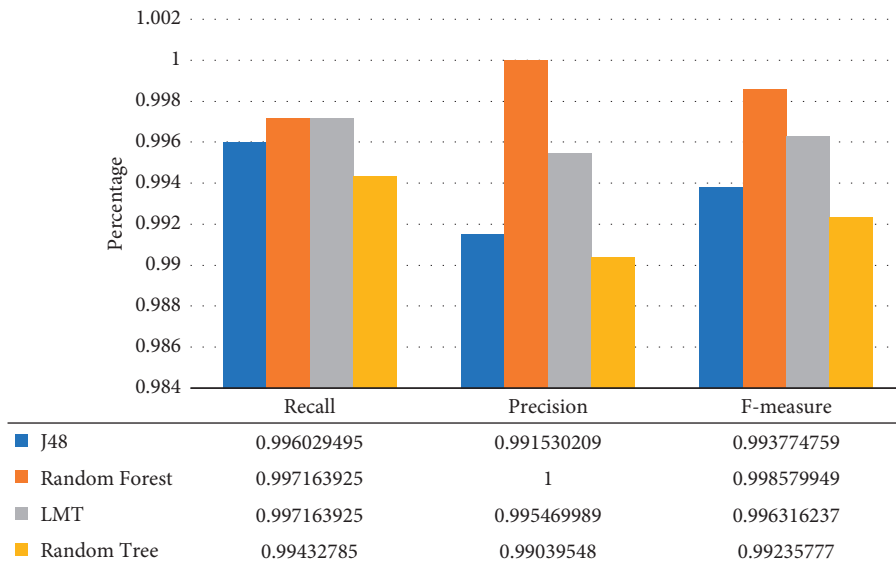


FIGURE 21: Classification results for top 60 features.

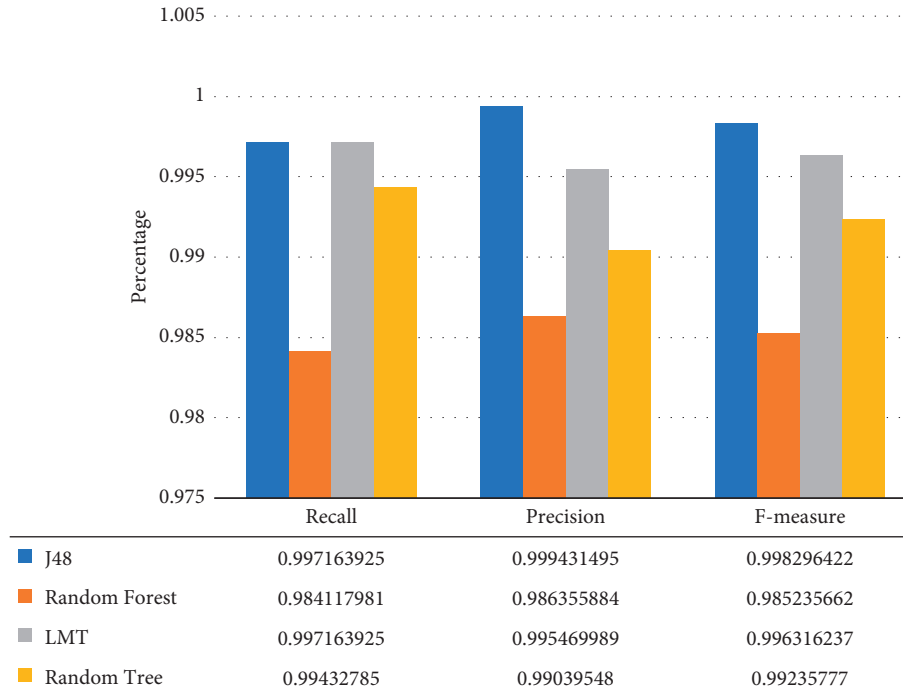


FIGURE 22: Classification results for top 80 features.

random forest classifier with precision value to 1. However, results also showed that recall value of random forest and LMT is exactly same, i.e., 0.997163.

4.4.4. Classification Results with Top 80 Dominant Features. A radical change in evaluation metric results is observed on considering 80 dominant features as shown in Figure 22. There is sudden rise in performance of J48; this is due to fact that J48 performs well when there are large numbers of features. Also, recall of J48 and LMT are found to be equivalent, i.e., 0.997163.

5. Conclusion and Future Work

Existing hybrid solutions majorly vary in feature set used for detection of Android ransomware. Most of the hybrid approaches focus on a specific ransomware family or a specific ransomware type or specific feature only. Those type-specific or family-specific solutions would be difficult to consider as a generalized solution. Extracting prominent features and feature selection methods is a research challenge. We used a total of 3249 applications samples to extract the static as well as dynamic features. The experimental results show that our proposed model is able to differentiate between clean and ransomware with improved precision. The results of our proposed hybrid framework outperform the existing static, dynamic, and hybrid approaches. Moreover, it also shows that the conglomeration of all dynamic features helps distinguish ransomware more effectively. Our proposed hybrid solution confirms accuracy of 99.85% with zero false positives while considering 60 prominent features. Further, it also justifies the feature selection algorithm used.

The considerable improvement in accuracy of our proposed hybrid framework encourages the use of the novel feature selection algorithm with ensemble machine learning classifiers also. We can also demonstrate the results over a larger dataset. In future, we may train ensemble learning models to detect as well as classify the ransomware into their families. Static features like URL, signatures, strings, and other resources and dynamic features like CPU usage and time could also help in achievement of promising results. Hence, we strongly recommend the use more static and dynamic features in future.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Disclosure

The presented work is PhD work of the author Tanya Gera.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

References

- [1] B. A. S. Al-rimy, M. A. Maarof, and S. Z. M. Shaid, "Ransomware threat success factors, taxonomy, and countermeasures: a survey and research directions," *Computers & Security*, vol. 74, pp. 144–166, 2018.
- [2] J. Song, C. Han, K. Wang, J. Zhao, R. Ranjan, and L. Wang, "An integrated static detection and analysis framework for

- android,” *Pervasive and Mobile Computing*, vol. 32, pp. 15–25, 2016.
- [3] N. Scaife, H. Carter, P. Traynor, and K. R. Butler, “Cryptolock (and drop it): stopping ransomware attacks on user data,” in *Proceedings of the 36th Int Conf Distrib Comput Syst.*, pp. 303–312, Nara, Japan, June 2016.
 - [4] F. Mercaldo, V. Nardone, A. Santone, and C. A. Visaggio, “Ransomware steals your phone. formal methods rescue it,” *Formal Techniques for Distributed Objects, Components, and Systems*, DisCoTec 2016, Heraklion, Crete, Greece, pp. 212–221, 2016.
 - [5] H. S. Galal, Y. B. Mahdy, and M. A. Atiea, “Behavior-based features model for malware detection,” *Journal of Computer Virology and Hacking Techniques*, vol. 12, no. 2, pp. 59–67, 2016.
 - [6] P. Wang and Y.-S. Wang, “Malware behavioural detection and vaccine development by using a support vector model classifier,” *Journal of Computer and System Sciences*, vol. 81, no. 6, pp. 1012–1026, 2015.
 - [7] P. Zhang and Y. Tan, “Hybrid concentration based feature extraction approach for malware detection,” in *Proceedings of the 28th IEEE Canadian Conference on Electrical and Computer Engineering*, pp. 140–145, Halifax, NS, USA, May 2015.
 - [8] L. Cen, C. S. Gates, L. Si, and N. Li, “A probabilistic discriminative model for android malware detection with decompiled source code,” *IEEE Transactions on Dependable and Secure Computing*, vol. 12, no. 4, pp. 400–412, 2015.
 - [9] G. Kapse, “Detection of malware on android based on application features,” *International Journal of Computer Science and Information Technology*, vol. 6, no. 4, pp. 3561–3564, 2015.
 - [10] N. Andronio, S. Zanero, and F. Maggi, “Heldroid: dissecting and detecting mobile ransomware,” in *Proceedings of the International Symposium on Recent Advances in Intrusion Detection*, pp. 382–404, Berlin, Heidelberg, October 2015.
 - [11] D. Maiorca, F. Mercaldo, G. Giacinto, C. A. Visaggio, and F. Martinelli, “R-PackDroid: API package-based characterization and detection of mobile ransomware,” in *Proceedings of the symposium on applied computing*, pp. 1718–1723, Marrakech, Morocco, March 2017.
 - [12] L. Casati and A. Visconti, “The dangers of rooting: data leakage detection in android applications,” *Mobile Information Systems*, vol. 2018, Article ID 6020461, 9 pages, 2018.
 - [13] G. Dini, F. Martinelli, I. Matteucci, M. Petrocchi, A. Saracino, and D. Sgandurra, “Risk analysis of android applications: a user-centric solution,” *Future Generation Computer Systems*, vol. 80, pp. 505–518, 2018.
 - [14] W. Wang, Y. Li, X. Wang, J. Liu, and X. Zhang, “Detecting Android malicious apps and categorizing benign apps with ensemble of classifiers,” *Future Generation Computer Systems*, vol. 78, pp. 987–994, 2018.
 - [15] A. Martín, V. Rodríguez-Fernández, and D. Camacho, “CANDYMAN: classifying android malware families by modelling dynamic traces with Markov chains,” *Engineering Applications of Artificial Intelligence*, vol. 74, pp. 121–133, 2018.
 - [16] J. Singh, D. Thakur, F. Ali, T. Gera, and K. S. Kwak, “Deep feature extraction and classification of android malware images,” *Sensors*, vol. 20, no. 24, p. 7013, 2020.
 - [17] T. Gera, J. Singh, D. Thakur, and P. Faruki, “A semi-automated approach for identification of trends in android ransomware literature,” in *Proceedings of the International Conference on Machine Learning for Networking*, pp. 265–283, Springer, Paris, France, November 2020.
 - [18] J. Singh, T. Gera, F. Ali, D. Thakur, K. Singh, and K.-s. Kwak, “Understanding research trends in android malware research using information modelling techniques,” *Computers, Materials & Continua*, vol. 66, no. 3, pp. 2655–2670, 2021, Available from: <http://www.techscience.com/cmcl/v66n3/41100>.
 - [19] M. Humayun, N. Jhanjhi, A. Alsayat, and V. Ponnusamy, “Internet of things and ransomware: evolution, mitigation and prevention,” *Egyptian Informatics Journal*, vol. 22, no. 1, pp. 105–117, 2021.
 - [20] M. Sun, X. Li, J. C. S. Lui, R. T. B. Ma, and Z. Liang, “Monet: a user-oriented behavior-based malware variants detection system for android,” *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 5, pp. 1103–1112, 2017.
 - [21] S. Arshad, M. A. Shah, A. Wahid, A. Mehmood, H. Song, and H. Yu, “SAMADroid: a novel 3-level hybrid malware detection model for android operating system,” *IEEE Access*, vol. 6, pp. 4321–4339, 2018.
 - [22] D. Arp, M. Spreitzenbarth, M. Hübner, H. Gascon, and K. Rieck, “Drebin: effective and explainable detection of android malware in your pocket,” *Proceedings 2014 Network and Distributed System Security Symposium*, vol. 2014, 2014 Available from, Article ID 23247.
 - [23] H. Han, “Identify and inspect libraries in android applications,” *Wirel Pers Commun*, Springer, US, pp. 1–13, 2018.
 - [24] S. Wang, Q. Yan, Z. Chen, B. Yang, C. Zhao, and M. Conti, “Detecting android malware leveraging text semantics of network flows,” *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 5, pp. 1096–1109, 2018.
 - [25] Developer Program Policy, Last accessed on 16 December. Available from: <https://support.google.com/googleplay/android-developer/answer/10286120?hl=en%0A>, 2020.
 - [26] F. Wei, Y. Li, S. Roy, X. Ou, and W. Zhou, “Deep ground truth analysis of current android malware,” in *Proceedings of the Detect Intrusions Malware, Vulnerability Assess 14th Int Conf DIMVA 2017*, pp. 252–276, Bonn, Ger, July 2017.
 - [27] Y. Fang, Y. Gao, F. Jing, and L. Zhang, “Android malware familial classification based on dex file section features,” *IEEE Access*, vol. 8, pp. 10614–10627, 2020.
 - [28] C. Li, K. Mills, D. Niu, R. Zhu, H. Zhang, and H. Kinawi, “Android malware detection based on factorization machine,” *IEEE Access*, vol. 7, pp. 184008–184019, 2019.
 - [29] S. Turker and A. B. Can, “AndMFC: android malware family classification framework,” in *Proceedings of the 2019 IEEE 30th International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC Workshops)*, pp. 1–6, Istanbul, Turkey, September 2019.
 - [30] R. Winsniewski, “Apktool: a tool for reverse engineering android apk files,” 2012, <http://ibotpeaches.github.io/Apktool/>.
 - [31] Y. Pan, X. Ge, C. Fang, and Y. Fan, “A systematic literature review of android malware detection using static analysis,” *IEEE Access*, vol. 8, pp. 116363–116379, 2020.
 - [32] HaboMalHunter, Last accessed on 10 April. Available from: <https://github.com/Tencent/%20HaboMalHunter>, 2019.
 - [33] A. Ferrante, M. Malek, F. Martinelli, F. Mercaldo, and J. Milosevic, “Extinguishing ransomware—a hybrid approach to android ransomware detection,” in *Proceedings of the International Symposium on Foundations and Practice of Security*, pp. 242–258, Springer, Montreal, QC, Canada, October 2017.
 - [34] A. Gharib and A. Ghorbani, “Dna-droid: a real-time android ransomware detection framework,” in *Proceedings of the*

- International Conference on Network and System Security*, Helsinki, Finland, September 2017.
- [35] F. Livingston, "Implementation of Breiman's random forest machine learning algorithm," *Mach Learn J Pap*, vol. 2005, pp. 1–13, 2005.
 - [36] J. R. Quinlan, *C4. 5: Programs for Machine Learning*. Elsevier, Amsterdam, Netherlands, 2014.
 - [37] A. Altaher, "An improved Android malware detection scheme based on an evolving hybrid neuro-fuzzy classifier (EHNFC) and permission-based features," *Neural Computing & Applications*, vol. 28, no. 12, pp. 4147–4157, 2016.
 - [38] M. Pal, "Random forest classifier for remote sensing classification," *International Journal of Remote Sensing*, vol. 26, no. 1, pp. 217–222, 2005.
 - [39] J. Ali, R. Khan, N. Ahmad, and I. Maqsood, "Random forests and decision trees," *Int J Comput Sci Issues*, vol. 9, no. 5, p. 272, 2012.

Research Article

Preventing Scan-Based Side-Channel Attacks by Scan Obfuscating with a Configurable Shift Register

Weizheng Wang , Yin Chen, Shuo Cai, and Yan Peng

School of Computer and Communication Engineering, Changsha University of Science and Technology, Changsha 410114, China

Correspondence should be addressed to Weizheng Wang; peakexpe@csust.edu.cn

Received 26 July 2021; Revised 12 October 2021; Accepted 23 October 2021; Published 5 November 2021

Academic Editor: Jie Cui

Copyright © 2021 Weizheng Wang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Scan test is widely used in integrated circuit test. However, the excellent observability and controllability provided by the scan test gives attackers an opportunity to obtain sensitive information by using scan design to threaten circuit security. Hence, the primary motivation of this paper is to improve the existing DFT technique, i.e., to enhance the chip security on the premise of guaranteeing test quality. In this paper, we propose a new scan design method against scan-based side-channel attack. In the proposed method, the encryption structure is adopted, which requires the correct test authorization code to carry out normal test operation. Without the correct test authorization, the attackers cannot obtain the desired scan data, preventing the scan-based side-channel attacks. Furthermore, the test authorization code is determined by the nonvolatile memory built into the chip to realize the inconsistency of the test authorization code for each chip.

1. Introduction

In recent years, several technologies, such as sensor networks [1–4], wireless communication [5–8], smart grid [9, 10], big data [11, 12], and internet of things [13, 14], have been developed rapidly and their security has been widely researched [15]. At the same time, the researcher has been paying more and more attention to the security issue of the underlying hardware [16–18].

In the manufacturing process of integrated circuit, defects are inevitable. When system intrinsic faults and faults in the integrated circuit occur simultaneously [19–21], fault detection will become more difficult [22–24]. In order to detect the faults of integrated circuit, testing is becoming an indispensable step and occupies an important position. Based on this, the design of scan chain to facilitate testing is proposed and widely used. Scan chain design can provide high controllability and observability during testing. However, the design of the scan chain gives attackers an open door while providing convenience. In [25], Yang et al. first proposed the scan-based side-channel attack. If the scan chain is not encrypted, sensitive information such as

intellectual property (IP) or secret keys [26, 27] could be exposed to attackers. Therefore, it is necessary to use a feasible solution to protect integrated circuits (ICs) from scan-based side-channel attacks [28].

In recent years, many scan-based attacks have been proposed to protect encryption systems. The scan-based side-channel attacks are mainly carried out through the acquisition and analysis of scan data. Currently, on-chip implementation of private key algorithms have been facing scan-based side-channel attacks, like Data Encryption Standard (DES) [29], Advanced Encryption Standard (AES) [30], Rivest-Shamir-Adleman (RSA) [31], Elliptic Curve Cryptography (ECC) [32], NTRUEcrypt [33], and Stream Cryptography based on Linear Feedback Shift Register (LFSR) [34].

Based on this, many countermeasures are put forward to counter the scan-based attacks [35–44]. Previously, the existing advanced DFT architecture includes test response compactor, X-masker [45, 46], and X-tolerance [47, 48]. They were regarded as a powerful countermeasure of resisting scan-based attacks. This DFT architecture makes it difficult to apply plaintext input and obtain intermediate

data from the scan chain, which provides a high level of security. However, recent research has shown that this strategy is also vulnerable. After inserting the test controller into the circuit under test, the state of the scan chain is cleared if the CUT is switched from functional mode to test mode [49]. This countermeasure is effective against mode-switching attacks, but they are not available against test-mode-only attacks. In [50], the technique keeps the password apart from the key module in test mode. It prevents an attacker from switching between test mode and functional mode. Another kind of methods obfuscate the scan output by changing the structure of the scan chain [51–56]. However, even without information about the scan cells, a skilled adversary can still carry out a signature attack [57, 58]. In [59], a solution is proposed, which is based on the lock and key of physical unclonable function, but this design method has a particularly high hardware overhead. Some methods resist scan-based attacks by reordering scan chains [60–69].

In order to protect the encryption chip from scan-based side-channel attacks, in this paper, we propose a new scan design method. In this method, only the user with the correct test authorization code can perform a normal scan. When a user without test authorization code tries to perform a scan test, the scan input/output data will be obfuscated. The test authorization code is determined by the values of the nonvolatile memory and the way the D flip-flops in a nonlinear shift register (NSR) connect with scan flip-flops. This means that the test authorization code for each encryption chip can be set differently. The main contributions of this paper are as follows:

- (1) A novel scan design scheme based on test authorization is presented to overcome scan attacks. By embedding a small management circuit, the enhanced DFT scheme improves significantly the security of chip. Furthermore, the proposed scheme does not incur significant performance penalties, for example, without decreasing the testability of the chip and increasing any timing delay.
- (2) The test authorization code can be changed when altering the configuration bits for the nonlinear shift register. Hence, the test authorization code can be different for two chips with the same design. This reduces substantially the risk of test authorization code disclosure. Even if one test authorization code is leaked, it will not affect all chips.

The rest of this paper is organized as follows. Section 2 describes the basic ideas, scan structure, and timing analysis of the proposed structure. Section 3 provides testability analysis, security analysis, and experimental results. Section 4 is the conclusion of this paper.

2. Proposed Secure Scan Design

2.1. Basic Idea of Proposed Secure Scan Design. In the proposed secure scan design, the test authorization code is used to manage scan operation. Only entering the correct test authorization code can enable the normal scan operation.

When the test authorization code is wrong, the scan-in stimulus and scan-out response are randomly XORed with the value of the node inside the combinational logic unit. At the same time, the wrong key will cyclically shift in the NSR, making data obfuscation elusory. Since the scan data is obfuscated, attackers will be misled into inferring incorrect results.

After power-on, the circuit is reset first. The operation mode of the circuit is controlled by the shift enable signal SE . When SE is set to low (“0”), the circuit enters in functional mode. When SE changes from “0” to “1,” enter the test authorization code from the first clock cycle of the scan test, and the N -bit test authorization code should be entered in N clock cycles. If the test authorization code is correct, normal scan operations can be carried out and the scan data will not be affected. If not, the circuit cannot perform the normal scan operation and the scan data will be obfuscated. The attacker will mistakenly believe that is the correct scan data and infer incorrect results. In order to strengthen the security of the encryption chip, the nonvolatile memory is used to control the test authorization code of each chip to be different. The test authorization code is determined by both the values of the nonvolatile memory and the output port (Q or \bar{Q}) of the D flip-flops in the NSR used to control the scan chain.

The proposed scan design method is a new architecture. In the following introduction, we first introduce the secure scan design and then show how to perform the test operation on a protected chip.

2.2. Scan Architecture of Proposed Secure Scan Design. As shown in Figure 1, the proposed secure scan structure is mainly composed of nonvolatile memory, nonlinear shift register (NSR), scan chain, and some control logic. The scan chain, made up of scan flip-flops (SFFs) marked in blue, is the intrinsic component in the standard scan design. The configurable NSR is used to store the test authorization code. If the test authorization code is N bits, an N -bit vector is needed to prestore in the nonvolatile memory to configure the NSR. The NSR contains $N D$ flip-flops, each of which is preceded by a 2-to-1 Multiplexer. The multiplexer has two data inputs, which are connected with the output Q and \bar{Q} of the front D flip-flop, respectively. The address input driven by a configuration bit in the nonvolatile memory is used to determine which data input is selected. Therefore, if the bit in the nonvolatile memory is “0,” it indicates that the output \bar{Q} of the front D flip-flop derives the next D flip-flop. Instead, if the bit is “1,” it implies the output Q of the front D flip-flop derives the next D flip-flop. It should be pointed out that the D input to the first D flip-flop is controlled by an additional 2-to-1 multiplexer. The two data inputs of the multiplexer are, respectively, connected to the last D flip-flop and the test authorization code input pin.

In the proposed structure, the scan chain is modified; that is, some XNOR gates are inserted between scan flip-flops. The output of a NAND gate serves as one of the inputs to the XNOR gate between scan flip-flops. The output Q (or its complement \bar{Q}) of a D flip-flop in NSR is connected with one input of the NAND gate, and the other input is driven by

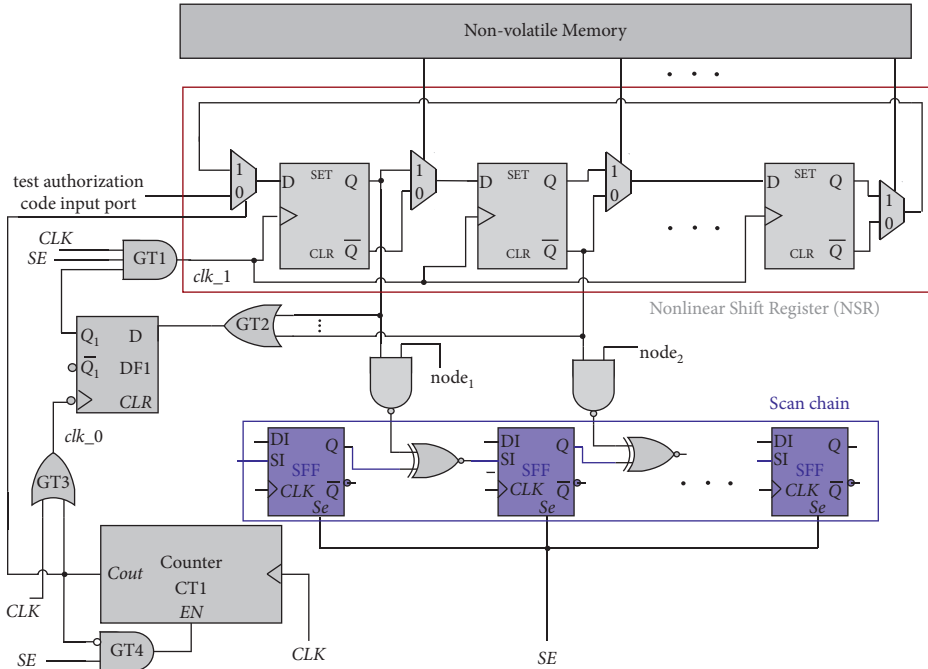


FIGURE 1: Proposed secure scan structure.

a combinational logic node selected randomly from CUT. On the assumption of Q connection, if the output Q of a NSR cell is 0, the NAND gate generates “1,” and the output of the XNOR gate is decided by the preceding scan flip-flop. Otherwise, if the output Q of a NSR cell is “1,” the output of the NAND gate is decided by the combinational logic node. When the combinational logic node is also “1,” the low level output of the NAND gate will make the succeeding scan flip-flop receive the opposite value of the preceding scan flip-flop. By this way, the logic obfuscation in the scan chain is achieved. Due to the uncertainty about the value of the combinational logic node, the logic obfuscation is haphazard and thus difficult to analyze. It is not difficult to see that if the Q output of a NSR cell is used to hardwire to the NAND gate, to enable the normal scan operation the state of the NSR cell should be 0. On the contrary, if it is the complement output \bar{Q} , the state of the NSR cell should be 1. We define the expected NSR state enabling the normal scan operation as the scan key. Meanwhile, the vector, which is loaded into NSR and used to generate the scan key, is defined as the test authorization code.

Besides being connected to the NAND gate, the Q output (or its complement \bar{Q}) of each NSR cell is also connected with an OR gate GT2. After the test authorization code is entered into the NSR completely, the output of the OR gate G2 can be latched into the D flip-flop DF1. The clock signal clk_0 of DF1 is driven by the OR gate GT3, which is controlled by the system clock CLK . The other input of GT3 is connected to the carry output $Cout$ of a module- N counter CT1. The clock signal clk_1 of the D flip-flop in NSR is driven by the output Q_1 of DF1 and the system clock CLK through the AND gate GT1. The enable signal of CT1 is marked as EN , which is connected to the complement of the carry output signal $cout$ through an AND gate GT4.

After the system reset or power-on, the module- N counter CT1 and DF1 will be initialized to zeros. The NSR is also initialized to all-zeros state.

In the test mode ($SE = 1$), when the output of the AND gate GT4 is high-level, EN port becomes high, and the module- N counter will be enabled. The module- N counter will start counting from zero. During this mode, test authorization code should be delivered first. When the correct test authorization code is entered completely, all the inputs of OR gate GT2 are “1” and the output of OR gate GT2 is “0,” so clk_1 will be “0.” Simultaneously, the counter reaches the maximum value of counting, so the carry output signal of CT1 becomes “1.” Due to the “1” value of carry output signal, the EN input of CT1 turns low, leading CT1 into the hold mode. The D flip-flop DF1 is locked because clk_0 is equal to “1” consistently. During this period, $Q_1 = 0$ and the output signal clk_1 of GT1 remains “0.” At this time, the D flip-flop in the NSR is locked by the clock clk_1 and the correct test authorization code is stored in the NSR until it is initialized. Because one input of the XNOR gate between SFFs is “1,” the scan data will not be affected and normal scan operations can be performed.

When the test authorization code is incorrect, that is, at least one bit is incorrect, the scan key will also be wrong. In this case, the output of the OR gate GT2 will be “1” after the module- N counter reaches the maximum value of counting. The “1” output of GT2 will be latched into DF1, the clock clk_0 of DF1 is disabled, and Q_1 remains “1.” Thus, the output clock clk_1 of GT1 will be active; that is, the shift operation in the NSR is enabled. The incorrect scan key will be shifted cyclically in NSR during the execution of the test operation. The shifted scan key will obfuscate the output of the scan chain through the XOR gate between SFFs. As a result, the attacker gets incorrect scan output, making the scan attack invalid.

As mentioned earlier, the test authorization code is determined by the combination of the values in the non-volatile memory and the connection style between NSR and the scan chain. The following is an example of inferring the test authorization code. Take a 5-bit test authorization code as an example. Assume that the value in nonvolatile memory is 01101, and the initial state in NSR after initialization is 00000. The test authorization code X_5, X_4, X_3, X_2, X_1 is delivered in five clock cycle from right to left. As can be seen from Table 1, after one cycle, the state of NSR becomes X_11001 . Eventually, after five cycles, the state of NSR is $X_5, \overline{X_4}, \overline{X_3}, \overline{X_2}, X_1$. The connection style between the D flip-flops in the NSR and the inserted NAND gates is shown in Figure 2. Thus, the expected scan key should be 11001. That is, $X_5, \overline{X_4}, \overline{X_3}, \overline{X_2}, X_1$ should be consistent with 11001. The right test authorization code can be solved, i.e., $X_5, X_4, X_3, X_2, X_1 = 10111$.

2.3. Timing Analysis of Proposed Secure Scan Design.

Assume that the state of the circuit before reset is unknown. The circuit is reset when the reset signal RST of the circuit changes from low to high. That is, all storage units are cleared to zero. In functional mode, RST is invalid and SE is low. In functional mode, NSR will not affect any operation of the circuit. Because the clk_1 is low, the NSR is disabled and the initial value of the NSR will not change. Low-level SE causes EN to be low. Based on this, the counter CT1 will not start counting, and the carry signal $cout$ remains "0." In summary, additional circuits will not work in functional mode.

In order to perform the test operation, SE should be set to "1," while clk_1 is activated. The N -bit test authorization code can be entered serially into the NSR input port. At this point, the EN port of CT1 is activated and the counter starts counting from "00." When the test authorization code is completely entered, the carry signal $cout$ of CT1 turns "1." The high value of $cout$ makes the enable signal EN of CT1 turn to "0," causing CT1 to be disabled. As described in Section 3, if the incorrect test authorization code is entered, the output Q1 of DF1 will be high due to the high output of GT2. The clock signal clk_1 of NSR is always consistent with CLK during test mode. The timing diagrams are illustrated in Figure 3. In this condition, incorrect test authorization code will shift bit by bit in the D flip-flop of NSR. That is, the scan data is the obfuscated data instead of the output data under scan test with correct test authorization code.

If the test authorization code entered is correct, the input of DF1 connected to the output of GT2 will be low. Because one input of GT1 is "0," the clock signal clk_1 of the D flip-flop in the NSR will be disabled, and the correct test authorization code is stored in the NSR. The timing diagrams are illustrated in Figure 4. In this condition, the scan test can be implemented normally.

3. Results and Performance Analysis

3.1. Testability Analysis. The insertion of security design does not affect the original testability of the circuit. All

commonly used testing techniques like stuck-at, and delay test can be applied. As long as the test authorization code is entered correctly, the normal scan operation can be performed, and the scan-out data will not be obfuscated.

Targeting at the stuck-at fault model, we do experiments on several big ITC'99 benchmark circuits including B17, B18, B19, B20, and B22. The results show, the fault coverage does not reduce for all these benchmark circuits with the same test set when the proposed secure scan design is integrated into them.

Since the added security design only adds logic gates, counters, and triggers, the faults occurring in the security scan design can be easily detected. When faults occur, although the test authorization code entered is correct, the output data will be still obfuscated. Then, the circuit will be treated as faulty one. Therefore, this does not affect the testability of the circuit.

3.2. Security Analysis. This section provides a detailed analysis of the security of the proposed structure by means of the following attack models.

3.2.1. Brute Force Attack. Since the test authorization code of the circuit is determined by the values in the nonvolatile memory and the way the NSR is connected with the scan chain, it is difficult to guess the test authorization code by brute force without obtaining specific design information about the circuit. The probability of randomly speculating the L -bit test authorization code to perform the scan test correctly is $(1/2)^L$. For $L = 64$, the probability of guessing the test authorization code is only 5.4×10^{-20} . In this case, it is impossible to obtain the test authorization code through brute force attack. In engineering applications, the attack probability and hardware overhead within the controllable range determine the value of L .

3.2.2. Differential Attack. Differential attack means that the attacker first runs in functional mode for several cycles and then switches to test mode to obtain an intermediate state [32]. Even if the attacker can dominate the scan chain through the primary input pins, the output data of scan chain will be obfuscated without the correct test authorization code. Therefore, the proposed secure scan structure can resist differential attack.

3.2.3. Test-Mode-Only Attack. Test-mode-only differential attack requires attackers to scan specific test vector pairs to obtain valuable information. However, in the proposed secure scan structure, these data will not be properly loaded into the scan chain due to the protection of obfuscation logic. In addition, incorrect keys can be cyclically shifted in the NSR during testing. Therefore, this leaves the obfuscated bits in an indeterminate state for each clock cycle while the scan operation is being performed. Therefore, the secure design proposed in this paper has the ability to resist test-mode-only attack.

TABLE 1: The state of example NSR.

0 th	0	0	0	0	0
1 st	X_1	1	0	0	1
2 nd	X_2	$\overline{X_1}$	1	0	1
3 rd	X_3	$\overline{X_2}$	$\overline{X_1}$	1	1
4 th	X_4	$\overline{X_3}$	$\overline{X_2}$	$\overline{X_1}$	0
5 th	X_5	$\overline{X_4}$	$\overline{X_3}$	$\overline{X_2}$	X_1
	1	1	0	0	1

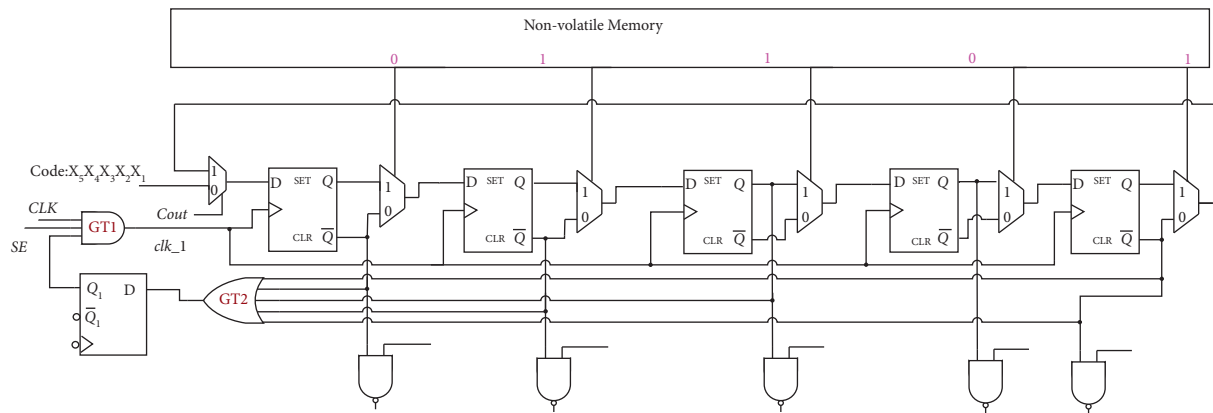


FIGURE 2: An example of inferring test authorization code.

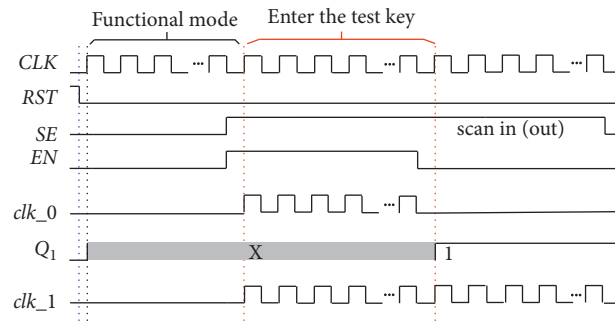


FIGURE 3: Timing diagram when the test authorization code is incorrect.

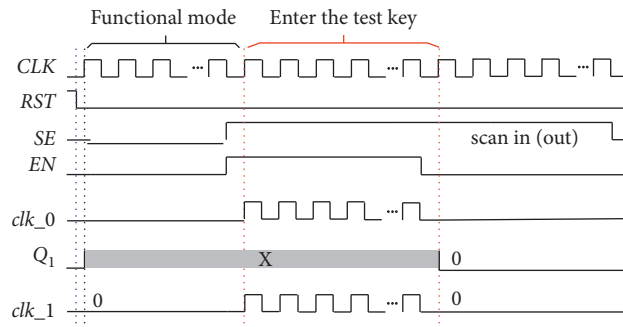


FIGURE 4: Timing diagram when the test authorization code is correct.

3.2.4. *Resetting Attack.* Resetting attack requires the attacker first resets the CUT, at which the state of all scan flip-flops is initialized to all-zeros. Then, the initial state is scanned with the given test authorization code. Finally, the attacker analyses the data from the scan-out result and determines whether the

test authorization code is correct. However, the secure scan design proposed in this paper has obfuscation characteristics. When the test authorization code is not correct, the wrong scan key shifted in the NSR, and the scan-out data will also be obfuscated. Hence, inferring the test authorization code bit by

TABLE 2: Area without inserting security design.

Area categories	Area
Combinational area	288499.656516
Buf/Inv area	82048.231581
Noncombinational area	63713.608595
Macro/black box area	0.000000
Net interconnect area	1611937.031250
Total cell area	352213.265112
Total area	1964150.296362

TABLE 3: Power consumption without inserting encryption design.

Internal power	Switching power	Leakage power	Total power
288.8334 uw	9.4736e+04 uw	8.9739e+06 nw	1.0400e+05 uw

TABLE 4: Area overhead with inserting encryption design.

Area categories	Area
Combinational area	288683.274511
Buf/Inv area	82261.829572
Noncombinational area	63958.268591
Macro/black box area	0.000000
Net interconnect area	1615052.812500
Total cell area	352641.543102
Total area	1967694.355602

TABLE 5: Power consumption with inserting encryption design.

Internal power	Switching power	Leakage power	Total power
311.7226 uw	1.0020e+05 uw	8.9633e+06 nw	1.0948e+05 uw

TABLE 6: Comparison of different secure scan design. Note: LOC denotes “launch-on-capture.”

Design	Area overhead (%)	Vulnerability	Probability of brute force	Test application
Proposed (64 bit authorization code)	0.18	None	2^{-64} (64 is the length of test authorization code)	All types of tests can be applied
MKR [30]	0.19	None	Brute force is inapplicable	Online testing cannot be applied
Mode reset [49]	~10	Test-mode-only attacks	Brute force is inapplicable	Online testing cannot be applied
Scan chain encryption [40]	2.92	Memory attack	2^{-m} (m is the length of test password)	All types of tests can be applied
FTSL-64 [59]	3.09	None	2^{-64}	Loc delay testing cannot be applied

bit from the scan-out data does not work. The proposed secure scan design can effectively resist the attacker using resetting attack to threaten the security of the circuit.

3.3. Overhead Analysis. In order to analyze area overhead, we perform experiments on AES circuit with Synopsys Design Compiler and Synopsys DFT Compiler. The area

without the security design is shown in Table 2, and the power consumption is shown in Table 3.

The area and power consumption after inserting the proposed secure scan design with 64-bit test authorization code are shown in Tables 4 and 5. By comparing the total area and total power consumption, it can be seen that the overhead and power consumption after inserting encryption design are well within the acceptable range.

Through the above analysis, the proposed secure scan design has high security and testability, as well as low area overhead and power consumption.

3.4. Overheads and Performance Comparison of Different Countermeasures. The area overhead and performance of the proposed secure scan design are compared with other countermeasures, MKR [30], Mode reset [49], scan chain encryption [40], and so on. The characteristics of these countermeasures are shown in Table 6. It can be seen from the comparison that the proposed secure scan design has many advantages, such as low area overhead, unscathed testing applications, and high security.

4. Conclusion

In this paper, a secure scan design is proposed to defeat the scan-based side-channel attacks. The proposed design adopts encryption structure, which requires the correct test authorization code to carry out normal test operation. The test authorization code needs to be inferred from both the configuration bit of a nonlinear shift register and the connection style between the nonlinear shift register and the scan chain. The configuration bits are stored in a nonvolatile memory, which can be configured arbitrarily by IP owner and are inaccessible for users and attackers. The proposed structure performs well in testability and security, and its overhead and power consumption are within acceptable range.

Data Availability

No data were used to support this study.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported in part by the Natural Science Foundation of Hunan Province under Grant nos. 2020JJ5604 and 2020JJ4622, the National Natural Science Foundation of China under Grant no. 61702052, and the Scientific Research Fund of Hunan Provincial Education Department under Grant no. 18A137.

References

- [1] J. Wang, Y. Gao, W. Liu, A. K. Sangaiah, and H.-J. Kim, "An intelligent data gathering schema with data fusion supported for mobile sink in wireless sensor networks," *International Journal of Distributed Sensor Networks*, vol. 15, no. 3, 2019.
- [2] B. Yin, S. Zhou, S. Zhang, K. Gu, and F. Yu, "On efficient processing of continuous reverse skyline queries in wireless sensor networks," *Ksii Transactions on Internet & Information Systems*, vol. 11, no. 4, pp. 1931–1953, 2017.
- [3] J. Wang, Y. Gao, X. Yin, F. Li, and H. Kim, "An enhanced PEGASIS algorithm with mobile sink support for wireless sensor networks," *Wireless Communications and Mobile Computing*, vol. 2018, Article ID 9472075, 2018.
- [4] J. Wang, Y. Gao, W. Liu, W. Wu, and S.-J. Lim, "An asynchronous clustering and mobile data gathering schema based on timer mechanism in wireless sensor networks," *Computers, Materials & Continua*, vol. 58, no. 3, pp. 711–725, 2019.
- [5] Y. Fei, L. Liu, L. Xiao, K. Li, and S. Cai, "A robust and fixed-time zeroing neural dynamics for computing time-variant nonlinear equation using a novel nonlinear activation function," *Neurocomputing*, vol. 350, pp. 108–116, 2019.
- [6] M. Long, Y. Chen, and F. Peng, "Simple and accurate analysis of BER performance for DCSK chaotic communication," *IEEE Communications Letters*, vol. 15, no. 11, pp. 1175–1177, 2011.
- [7] Y. Fei, L. Gao, L. Liu, S. Qian, S. Cai, and Y. Song, "A 1 V, 0.53 ns, 59 μ W current comparator using standard 0.18 μ m CMOS technology," *Wireless Personal Communications*, vol. 111, pp. 843–851, 2020.
- [8] Y. Fei, Q. Tang, W. Wang, and H. Wu, "A 2.7 GHz low-phase-noise LC-QVCO using the gate-modulated coupling technique," *Wireless Personal Communications*, vol. 86, no. 2, pp. 671–681, 2016.
- [9] Q. Tang, K. Yang, D. Zhou, Y. Luo, and F. Yu, "A real-time dynamic pricing algorithm for smart grid with unstable energy providers and malicious users," *IEEE Internet of Things Journal*, vol. 3, no. 4, pp. 554–562, 2016.
- [10] T. Qiang, M. Xie, K. Yang, Y. Luo, D. Zhou, and Y. Song, "A decision function based smart charging and discharging strategy for electric vehicle in smart grid," *Mobile Networks and Applications*, vol. 24, pp. 1722–1731, 2019.
- [11] J. Wang, Y. Yang, T. Wang, R. Sherratt, and J. Zhang, "Big data service architecture: a survey," *Journal of Internet Technology, [S.l.]*, vol. 21, no. 2, pp. 393–405, 2020.
- [12] J. Wang, Y. Yang, J. Zhang, X. Yu, O. Alfarraj, and A. Tolba, "A data-aware remote procedure call method for big data systems," *Computer Systems Science and Engineering*, vol. 35, no. 6, pp. 523–532, 2020.
- [13] B. Yin and X. Wei, "Communication-Efficient data aggregation tree construction for complex queries in IoT applications," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 3352–3363, 2019.
- [14] W. Li, Z. Chen, X. Gao, W. Liu, and J. Wang, "Multimodel framework for indoor localization under mobile edge computing environment," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 4844–4853, 2019.
- [15] M. Long, F. Peng, and H.-y. Li, "Separable reversible data hiding and encryption for HEVC video," *Journal of Real-Time Image Processing*, vol. 14, no. 1, pp. 171–182, 2018.
- [16] J. Zhang, C. Shen, H. Su, M. T. Arafain, and G. Qu, "Voltage over-scaling-based lightweight Authentication for IoT security," *IEEE Transactions on Computers*, pp. 1–14, 2021.
- [17] W. Wang, X. Wang, J. Wang, N. N. Xiong, S. Cai, and P. Liu, "Ensuring Cryptography chips security by preventing scan-based side-channel attacks with improved DFT architecture," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, pp. 1–15, 2021.
- [18] J. Zhang and G. Qu, "Physical unclonable function-based key sharing via machine learning for IoT security," *IEEE Transactions on Industrial Electronics*, vol. 67, no. 8, pp. 7025–7033, 2020.
- [19] X. He, Z. Wang, L. Qin, and D. Zhou, "Active fault-tolerant control for an Internet-based networked three-tank system," *IEEE Transactions on Control Systems Technology*, vol. 24, no. 6, pp. 2150–2157, 2016.

- [20] Y.-Y. Wang, Y. Sun, C.-F. Chang, and Y. Hu, "Model-based fault detection and fault-tolerant control of SCR urea injection systems," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 6, pp. 4645–4654, 2016.
- [21] H. Badihi, Y. Youmin Zhang, and H. Hong, "Wind turbine fault diagnosis and fault-tolerant torque load control against actuator faults," *IEEE Transactions on Control Systems Technology*, vol. 23, no. 4, pp. 1351–1372, 2015.
- [22] I. Pomeranz, "On the computation of common test data for broadside and skewed-load tests," *IEEE Transactions on Computers*, vol. 61, no. 4, pp. 578–583, 2012.
- [23] I. Pomeranz, "Multicycle tests with constant primary input vectors for increased fault coverage," *IEEE Trans. CAD Integrated Circuit System*, vol. 31, no. 9, pp. 1428–1438, 2018.
- [24] S. Zhang, K. R. Pattipati, Z. Hu, and X. Wen, "Optimal selection of imperfect tests for fault detection and isolation," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 43, no. 6, pp. 1370–1384, 2013.
- [25] B. Yang, K. Wu, and R. Karri, "Scan based side channel attack on dedicated hardware implementations of data encryption standard," in *Proceedings of the International Test Conference*, pp. 339–344, Charlotte, NC, USA, October 2004.
- [26] S. S. Ali, O. Sinanoglu, and R. Karri, "Test-mode-only scan attack using the boundary scan chain," in *Proceedings of the 19th IEEE Europe Test Symposium (ETS)*, pp. 1–6, Paderborn, Germany, May 2014.
- [27] M. Tehranipoor and C. Wang, *Introduction to Hardware Security and Trust*, Springer, New York, NY, USA, 2011.
- [28] J. Dworak and A. Crouch, "A call to action: securing IEEE 1687 and the need for an IEEE test security standard," in *Proceedings of the IEEE 33rd VLSI Test Symposium (VTS)*, pp. 1–4, Napa, CA, USA, April 2015.
- [29] B. Yang, K. Wu, and R. Karri, "Scan based side channel attack on dedicated hardware implementations of data encryption standard," in *Proceedings of the International Conference Test*, pp. 339–344, Boston, MA, USA, October 2004.
- [30] B. Yang, K. Wu, and R. Karri, "Secure scan: a design-for-test architecture for crypto chips," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 25, no. 10, pp. 2287–2293, 2006.
- [31] R. Nara, K. Satoh, M. Yanagisawa, T. Ohtsuki, and N. Togawa, "Scan-based side-channel attack against RSA cryptosystems using scan signatures," *IEICE - Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. E93-A, no. 12, pp. 2481–2489, 2010.
- [32] J. D. Rolt, G. Di Natale, M.-L. Flottes, and B. Rouzeyre, "A novel differential scan attack on advanced DFT structures," *ACM Transactions on Design Automation of Electronic Systems*, vol. 18, no. 4, 2013.
- [33] A. A. Kamal and A. M. Youssef, "A scan-based side channel attack on the NTRUEncrypt cryptosystem," in *Proceedings of the 7th International Conference Availability, Reliability Security*, pp. 402–409, Prague, Czech Republic, August 2012.
- [34] Y. Liu, K. Wu, and R. Karri, "Scan-based attacks on linear feedback shift register based stream ciphers," *ACM Transactions on Design Automation of Electronic Systems*, vol. 16, no. 2, 2011.
- [35] J. Da Rolt, A. Das, G. Di Natale, M. L. Flottes, B. Rouzeyre, and I. Verbauwhede, "A scan-based attack on elliptic curve cryptosystems in presence of industrial design-for-testability structures," in *Proceedings of the IEEE International Symposium on Defect and Fault Tolerance in VLSI Systems*, pp. 43–48, Austin, TX, USA, October 2012.
- [36] J. Da Rolt, G. Di Natale, M. Flottes, and B. Rouzeyre, "New security threats against chips containing scan chain structures," in *Proceedings of the 2011 IEEE International Symposium on Hardware-Oriented Security and Trust*, p. 110, San Diego, CA, USA, June 2011.
- [37] J. Da Rolt, G. Di Natale, M. Flottes, and B. Rouzeyre, "Are advanced DFT structures sufficient for preventing scan-attacks?" in *Proceedings of the 2012 IEEE 30th VLSI Test Symposium (VTS)*, pp. 246–251, Hyatt Maui, HI, USA, April 2012.
- [38] S. S. Ali, S. M. Saeed, O. Sinanoglu, and R. Karri, "Novel test-mode-only scan attack and countermeasure for compression-based scan architectures," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 34, no. 5, pp. 808–821, 2015.
- [39] G.-M. Chiu and J. C.-M. Li, "A secure test wrapper design against internal and boundary scan attacks for embedded cores," *IEEE Transactions on Very Large Scale Integration Systems*, vol. 20, no. 1, pp. 126–134, 2012.
- [40] M. Da Silva, M.-L. Flottes, G. Di Natale, and B. Rouzeyre, "Preventing scan attacks on secure circuits through scan chain encryption," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 38, no. 3, pp. 538–550, 2019.
- [41] M. Da Silva, M. I. Flottes, G. Di Natale, B. Rouzeyre, P. Prinetto, and M. Restifo, "Scan chain encryption for the test, diagnosis and debug of secure circuits," in *Proceedings of the 2017 22nd IEEE European Test Symposium (ETS)*, pp. 1–6, Limassol, Cyprus, May 2017.
- [42] M. Da Silva, E. Valea, M. L. Flottes, S. Dupuis, G. Di Natale, and B. Rouzeyre, "A new secure stream cipher for scan chain encryption," in *Proceedings of the 2018 IEEE 3rd International Verification and Security Workshop (IVSW)*, Platja d'Aro, Spain, October 2018.
- [43] P. Raiola, M. Kochte, A. Atteya et al., "Detecting and Resolving Security violations in reconfigurable scan networks," in *Proceedings of the 2018 24th IEEE International Symposium on On-Line Testing and Robust Design (IOLTS)*, Platja d'Aro, Spain, October 2018.
- [44] G. Sengar, D. Mukhopadhyay, and D. R. Chowdhury, "Secured flipped scan-chain model for crypto-architecture," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 26, no. 11, pp. 2080–2084, 2007.
- [45] O. Novak, J. Jenicek, and M. Rozkovec, "Sequential test decompressors with fast variable wide spreading," in *Proceedings of the 19th IEEE Design Diagnostics Electron. Circuits System Symposium*, pp. 132–137, Kosice, Slovakia, April 2016.
- [46] J.-H. Kang, N. A. Toubia, and J.-S. Yang, "Reducing control bit overhead for X-masking/X-canceling hybrid architecture via pattern partitioning," in *Proceedings of the 53rd Design Automation Conference*, pp. 344–349, Austin, TX, USA, June 2016.
- [47] J. Da Rolt, G. Di Natale, M.-L. Flottes, and B. Rouzeyre, "Scan attacks and countermeasures in presence of scan response compactors," in *Proceedings of the 16th IEEE European Test Symposium (ETS)*, pp. 19–24, Trondheim, Norway, May 2011.
- [48] A. Das, B. Ege, S. Ghosh, L. Batina, and I. Verbauwhede, "Security analysis of industrial test compression schemes," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 32, no. 12, pp. 1966–1977, 2013.
- [49] D. Hély, F. Bancel, M.-L. Flottes, and B. Rouzeyre, "Securing scan control in crypto chips," *Journal of Electronic Testing*, vol. 23, no. 5, pp. 457–464, 2007.

- [50] W. Wang, J. Wang, W. Wang, P. Liu, and S. Cai, "A secure DFT architecture protecting crypto chips against scan-based attacks," *IEEE Access*, vol. 7, pp. 22206–22213, 2019.
- [51] J. Lee, M. Tehranipoor, C. Patel, and J. Plusquellic, "Securing designs against scan-based side-channel attacks," *IEEE Transactions on Dependable and Secure Computing*, vol. 4, no. 4, pp. 325–336, 2007.
- [52] Y. Atobe, Y. Shi, M. Yanagisawa, and N. Togawa, "Secure scan design with dynamically configurable connection," in *Proceedings of the 19th IEEE Pacific Rim International Symposium Dependable Computing (PRDC)*, pp. 256–262, Vancouver, Canada, December 2013.
- [53] A. Cui, Y. Luo, and C.-H. Chang, "Static and dynamic obfuscations of scan data against scan-based side-channel attacks," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 2, pp. 363–376, 2017.
- [54] Y. Atobe, Y. Shi, M. Yanagisawa, and N. Togawa, "Dynamically changeable secure scan architecture against scan-based side channel attack," in *Proceedings of the International SoC Design Conference*, pp. 155–158, Jeju Island, South Korea, November 2012.
- [55] D. Zhang, M. He, X. Wang, and M. Tehranipoor, "Dynamically obfuscated scan for protecting IPs against scan-based attacks throughout supply chain," in *Proceedings of the 35th IEEE VLSI Test Symposium*, pp. 141–146, Las Vegas, NV, USA, April 2017.
- [56] X. Wang, D. Zhang, M. He, D. Su, and M. Tehranipoor, "Secure scan and test using obfuscation throughout supply chain," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 37, no. 9, pp. 1867–1880, 2018.
- [57] H. Kodera, M. Yanagisawa, and N. Togawa, "Scan-based attack against DES cryptosystems using scan signatures," in *Proceedings of the IEEE Asia Pacific Conference Circuits System*, pp. 599–602, Kaohsiung, Taiwan, December 2012.
- [58] R. Nara, N. Togawa, M. Yanagisawa, and T. Ohtsuki, "A scan-based attack based on discriminators for AES cryptosystems," *IEICE - Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. 12, no. 12, pp. 3229–3237, 2009.
- [59] A. Cui, C.-H. Chang, W. Zhou, and Y. Zheng, "A new PUF based lock and key solution for secure in-field testing of cryptographic chips," *IEEE Transactions on Emerging Topics in Computing*, vol. 9, no. 2, pp. 1095–1105, 2021.
- [60] U. Chandran and D. Zhao, "SSKTC a high-testability low-overhead scan architecture with multi-level security integration," in *Proceedings of the 27th IEEE VLSI Test Symposium (VTS)*, pp. 321–326, Santa Cruz, CA, USA, May 2009.
- [61] M. A. Razzaq, V. Singh, and A. Singh, "SSTKR secure and testable scan design through test key randomization," in *Proceedings of the 20th IEEE Asian Test Symp. (ATS)*, pp. 60–65, New Delhi, India, November 2011.
- [62] S. Paul, R. S. Chakraborty, and S. Bhunia, "VIm-scan: a low overhead scan design approach for protection of secret key in scan-based secure chips," in *Proceedings of the 25th IEEE VLSI Test Symposium*, pp. 455–460, Berkeley, CA, USA, May 2007.
- [63] Y. Luo, A. Cui, G. Qu, and H. Li, "A new countermeasure against scan-based side-channel attacks," in *Proceedings of the IEEE International Symposium Circuits System (ISCAS)*, pp. 1722–1725, Montreal, Canada, May 2016.
- [64] G. Sengar, D. Mukhopadhyay, and D. R. Chowdhury, "An efficient approach to develop secure scan tree for crypto-hardware," in *Proceedings of the International Conference Advanced Computer Communication (ADCOM)*, pp. 21–26, Guwahati, India, January 2007.
- [65] Y. Atobe, S. Youhua, M. Yanagisawa, and N. Togawa, "State dependent scan flip-flop with key-based configuration against scan-based side-channel attack on RSA circuit," in *Proceedings of the Asia Pacific Conference Circuits System*, pp. 607–610, Kaohsiung, Taiwan, December 2012.
- [66] S. Paul, R. S. Chakraborty, and S. Bhunia, "VIm-scan: a low overhead scan design approach for protection of secret key in scan-based secure chips," in *Proceedings of the 25th IEEE VLSI Test Symposium (VTS'07)VTS*, pp. 455–460, Berkeley, CA, USA, May 2007.
- [67] L. Pierce and S. Tragoudas, "Enhanced secure architecture for joint action test group systems," *IEEE Transactions on Very Large Scale Integration Systems*, vol. 21, no. 7, pp. 1342–1345, 2013.
- [68] R. Nara, N. Togawa, M. Yanagisawa, and T. Ohtsuki, "Scan-based attack against elliptic curve cryptosystems," in *Proceedings of the Asia South Pacific Design Automation Conference (ASP-DAC)*, pp. 407–412, Taipei, Taiwan, January 2010.
- [69] M. Fujishiro, M. Yanagisawa, and N. Togawa, "Scan-based attack against Trivium stream cipher independent of scan structure," in *Proceedings of the IEEE 10th International Conference ASIC (ASICON)*, pp. 1–4, Shenzhen, China, October 2013.

Research Article

Blockchain-Enabled Intelligent Video Caching and Transcoding in Clustered MEC Networks

Yan Li ^{1,2} and Zheng Wan ¹

¹School of Information Management, Jiangxi University of Finance and Economics, No. 665, West Yuping Road, Nanchang, Jiangxi 330032, China

²Nanchang Institute of Technology, No. 289, Tianxiang Road, Nanchang, Jiangxi 330099, China

Correspondence should be addressed to Zheng Wan; wanzheng97@163.com

Received 23 May 2021; Revised 15 July 2021; Accepted 16 August 2021; Published 8 September 2021

Academic Editor: Ke Gu

Copyright © 2021 Yan Li and Zheng Wan. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In recent years, the number of smart devices has exploded, leading to an unprecedented increase in demand for video live and video-on-demand (VoD) services. Also, the privacy of video providers and requesters and the security of requested video data are much more threatened. In order to solve these issues, in this paper, a blockchain-enabled CMEC video transmission model (Bl-CMEC) for intelligent video caching and transcoding will be proposed to ensure the transactions' transparency, system security, user information privacy, and integrity of the video data, enhance the ability of servers in actively caching popular video content in the CMEC system, and realize transcoding function at network edge nodes. Furthermore, we chose a scheme based on deep reinforcement learning (DRL) to intelligently access the intracluster joint caching and transcoding decisions. Then, the joint video caching and transcoding decision smart contract is specially designed to automatically manage the transaction process of the joint caching and transcoding service, which records key information of joint caching and transcoding transactions and payment information on a continuous blockchain. The simulation results demonstrate that the proposed Bl-CMEC framework not only can provide users with better QoE performance for video streaming service but also can ensure the security, integrity, and consistency for the video providers, video requesters, and video data.

1. Introduction

People are becoming more and more dependent on network services, especially during the period of COVID-19 pandemic, many activities and works are carried out on the network. Furthermore, the most important network service is the video streaming service. In recent years, because the number of smart devices has exploded, there is an increasing demand for video live and video-on-demand (VoD) services. In video streaming services, higher data rates and larger system capacity are usually required to meet the ever-increasing users' needs, which has become a more challenging task. According to the summary of the Cisco Visual Network Index [1], mobile smart devices' videos compose around more than 50% of the total data traffic. And, it is expected to grow further to about 79% of the total data traffic in 2022.

Because of the huge demand for mobile smart devices' videos, the operators of mobile networks are not able to meet the users' demand for high quality of experience (QoE) in video live and VoD services.

In order to solve this issue, the proposal of mobile edge computing (MEC) has brought new opportunities for the optimization of wireless video transmission [2–7]. Utilizing the communication, caching, computing, and control (4C) capabilities of edge devices, it could provide proactive video caching, transcoding, and distribution services at the network edge in the mobile networks. Also, it could reduce the burden of the backbone network and improve the video quality of experience for requested users.

At the same time, smart device users may have different needs for specific videos because of the heterogeneity of smart device users' caching and computing capabilities and

changes in network conditions. For example, the smart device users with better network always ask for high QoE videos, while the smart device users with poor network generally prefer the videos with appropriate QoE. To reduce the computational load of the cloud center and transmission cost at different formats and versions of videos in the backbone network, the Content Distribution Network (CDN) is proposed which may only push a certain format and version of the video stream to the network edge and requires intelligent transcoding and distribution for tasks, as well as adaptive allocation of network resources. Especially, the cooperative transcoding decision and task assignment are needed to decide which edge nodes carry out cooperative transcoding and what kind of transcoding task is assigned to each node. The influencing factors of video transcoding and caching decisions include video content popularity, user demand and distribution, the capabilities of each edge node, and bandwidth resources between nodes. Because of this issue, adaptive bitrate streaming (ABR) [8] has been widely proposed to improve the QoE of video data which serve for smart device users in the Internet.

In addition, the blockchain has developed rapidly in recent years, which is a new fashion application mode. Its core content includes P2P transmission, encryption algorithm, distributed data storage, and consensus mechanism [9, 10], and the consensus mechanism is the most important content in blockchain. Blockchain has acted as a very effective distributed management framework which has been widely used in many fields. Through the blockchain module integrated under the framework of MEC, the data resource security protection and monitoring can be realized for the processing of video data at network edge.

In this paper, a blockchain-enabled framework for Clustered Mobile Edge Computing (CMEC) system is proposed, which can integrate the MEC networks and the CDN networks by setting the CDN tips. The experiments on comparison of QoE and bandwidth cost between CMEC and other schemes have been performed in our own previous paper [11]. The CMEC method can promote intracenter collaboration among the MEC nodes in one cluster. So, it can reduce the additional processing costs and backhaul consumption. Furthermore, the proposed blockchain-enabled CMEC-based video transmission model in this paper can seamlessly enable the blockchain scheme into our Clustered MEC network, connecting with the popular CDN video transmission system. Then, we deploy the blockchain structure into the CMEC system, which can be set at CDN tips or edge clusters. In the proposed scheme, the CDN tip and some edge nodes make the network edge area in the local network area. Then, the proposed model optimizes the entire network transmission of wireless video data by using the collaborative capabilities of edge cluster in communication, caching, computing, and control (4C). Specifically, the main contributions of this paper can be summarized as follows:

- (i) Blockchain-enabled CMEC-based video transmission model (Bl-CMEC): a video transmission system framework model with incorporating

blockchain technology is designed to actively cache popular video content in the CMEC system and realize transcoding function at network edge nodes. This model is used to improve the allocation of video caching resources and computing resources in edge cluster nodes and also to optimize user QoE from the perspective of mobile users.

- (ii) Blockchain empowerment: the joint caching and transcoding transactions between network edge node and smart device users are implemented in blockchain by a smart contract. The smart contract is specially designed to manage the transaction process of the joint caching and transcoding service, which records the joint caching and transcoding transaction and payment information on a continuous blockchain. Furthermore, the smart contract can check the integrity of results returned from the network edge node to achieve adaptive video transmission optimization and make security protection of video data in a blockchain-enabled CMEC-based environment.
- (iii) Intelligent scheme design: it can intelligently obtain and implement the decision of allocating the caching and computing resources at the network edge node in one cluster. It has two sections, namely, DQN-Based Video Caching and Transcoding Algorithm (DQN-VCT) (section 1) and Implementation of Video Caching and Transcoding Decision Smart Contract (section 2). According to mobile users' demand changes and network time-varying conditions, each network edge node determines the optimal caching and transcoding price. Then, we use the deep reinforcement learning-based algorithm to acquire the smart device users' optimal caching and transcoding decision. The decision is to meet the best QoE needs of smart device users and obtain better video services.
- (iv) Sufficient performance evaluation: because of extensive and sufficient simulations, we analyze the performance of the proposed intelligent video caching and transcoding scheme in this paper based on blockchain-enabled CMEC system environment.

This paper is organized as follows. Section 2 presents related work. The system model design is described in Section 3. Furthermore, Section 4 formulates the blockchain empowerment mode. Problem formation and intelligent video caching and transcoding scheme is introduced in Section 5. The analysis of simulation experiments is given in Section 6. Lastly, the conclusions are given in Section 7.

2. Related Work

In recent years, the number of smart devices has exploded, leading to an unprecedented increase in demand for video live and VoD services. Also, the privacy of video providers and requesters and the security of requested video data are much more threatened in the mobile edge computing

system. Thus, the blockchain technology can help to enable the privacy of video providers and the security of requested video data. Through the distributed storage mechanism of the blockchain, the security of video information is improved, the collection of individual video data is realized through the personal ledger of blockchain, and the authorized use of video data is realized through asymmetric encryption and public and private key design in blockchain [12–15]. It is essential to introduce the blockchain into the adaptive video services in the mobile edge computing system.

The integration of blockchain and MEC to solve the corresponding practical problems is currently a hot spot for many scholars, which is a very promising development direction [12, 16–19]. On the one hand, MEC nodes in network edge can provide a low-latency, much more convenient and distributed computing offloading scheme for smart mobile devices with only limited resources. Edge computing devices have powerful computing and storage capabilities compared to general user mobile devices, while blockchain services require powerful computing capabilities. Therefore, MEC provides the possibility for mobile users to enjoy blockchain services. On the other hand, blockchain can be used as an auxiliary framework to manage the provision of mobile edge computing resources and turn the supply of edge computing resources into a blockchain application. This not only enhances the security of MEC resources but also regulates the occupation and purchase of edge computing resources.

Based on the literature on the combination of mobile edge computing and blockchain, the recent research mainly focuses on two aspects. First, such research about security and privacy protection mainly lies in how to introduce the blockchain technology into MEC system to achieve the security and safety of cached content. Furthermore, MEC-enabled blockchain-based distributed video system architecture is used to solve the problem about the allocation of decentralized resource for video caching, transcoding, and delivery at blockchain-based video streaming system.

2.1. Blockchain-Enabled MEC-Based Caching Strategy. In the process of task offloading, the transmitted information is vulnerable to attacks, resulting in incomplete data. In view of this challenge, Xu et al. [20] proposed a blockchain-enabled computing offloading method which is called BeCome. In BeCome, to ensure the data integrity, they introduced the blockchain technology into the edge computing scheme. To address the issues of data security and users privacy, Feng et al. [21] adopted blockchain technology to ensure the reliability and irreversibility of cache data in the MEC network system. Also, they developed a framework of cooperative computing sharing and resource allocation for the blockchain-enabled MEC-based network system. Guo et al. [22] proposed a blockchain-enabled MEC-based framework for adaptive computing offloading and resource allocation in the future wireless networks. In this method, blockchain is used to provide management and control function. Then, the problem acted as a joint optimization issue and deep

reinforcement learning based methods are adopted to address this problem. Because of the possession of very sensitive personal information, the vehicle may be unwilling to cache its content to an untrusted cache provider. Dai et al. [23] integrated DRL and permissioned blockchain into the vehicle network, so as to acquire a secure and smart content caching method, by which a distributed content caching framework based on blockchain has been proposed. Content caching in Mobile Cyber-Physical System also faces some security issues. To address these issues, Xu et al. [24] proposed a new blockchain-based trusted network edge caching solution for mobile smart device users in a Mobile Cyber-Physical System.

Applying blockchain to MEC cache system mainly solves the safety problem of cache content, MEC servers, mobile equipment users, etc. In terms of blockchain-enabled MEC-based systems, on the one hand, such research mainly lies in the use of blockchain technology to realize the privacy protection and security of cached content. On the other hand, blockchain technology can be adopted to solve data integrity issues in edge computing and to monitor the resources of edge computing devices.

2.2. MEC-Enabled Blockchain-Based Distributed Video Delivery Strategy. In the blockchain-based video delivery network system, in order to meet the different needs of smart device users, a lot of computing resources are needed to transcoding them into different versions and formats for the heterogeneous quality and format of video streams. To solve this problem, Liu et al. [25] and Liu et al. [26] have proposed a MEC-enabled blockchain-based architecture using MEC technology, with a series of smart contracts that can acquire self-organization at video transcoding and delivery services, especially without a centralized controller. Some emerging video streaming platforms want to build cryptocurrency-based payment systems and p2p content distribution architectures by using the blockchain technology. Liu et al. [27] proposed a novel transcoding framework that supports the MEC network for blockchain-based video delivery scheme, designing an adaptive block size mode for the underlying blockchain. Furthermore, Zhang et al. [28, 29] proposed an incentive mechanism for blockchain-based cache and delivery systems. By this incentive mechanism, the willingness of both MEC network cache nodes and D2D can be guaranteed by meeting their expected rewards for cache sharing. The existing offloading methods based on DRL always suffer from a slow convergence which is caused by the high-dimensional action spaces. Qiu et al. [30] presented a new free-model DRL online computing offloading mode. This method is used to solve the computing offloading of data processing tasks and mining tasks in wireless blockchain networks. Effective computing diversion cannot be achieved in MEC with blockchain because mobile devices do not always have enough tokens to bear the cost of diversion services. Zhang et al. [31] analyzed the combined computing offloading and coin loan problems of blockchain-empowered MEC to optimize the total cost of all smart mobile equipment.

Such research mainly focuses on two aspects. The first aspect is to use mobile edge computing to solve the mining tasks and data processing tasks in the wireless blockchain network. In addition, in the blockchain-based video system, it works to implement the distributed resource allocation issues for video transcoding and delivery.

2.3. Motivation. Because of the high dynamics of the MEC network system, the data security and privacy protection of network edge service providers are a major challenge. Blockchain technology can construct a decentralized and secure resource sharing scheme, while Artificial Intelligence (AI) can explore and solve issues with time-varying, uncertain and complex characteristics [32–36]. The blockchain technology and MEC system both have the same decentralized characteristics, making their combination natural. Motivated by recent research results, blockchain technology can be introduced into the MEC system to support many management and security services in mobile edge computing. Also, DRL-based video transmission strategy in the MEC environment is extensively studied recently [37–43]. The DRL method can jointly solve the problems of cache content location decision, cache update strategy, and cache content delivery. Deep reinforcement learning is used to analyze and learn network information through deep learning, so as to use reinforcement learning to achieve resource scheduling.

Based on the related work, applying blockchain to MEC cache system mainly addresses the safety threat of the cached content, MEC servers, mobile equipment users, etc. In terms of blockchain-enabled MEC-based systems, such research only lies in the use of blockchain technology at the caching strategy, which realizes the security and privacy protection of the cached content. Furthermore, pursuing for the new distributed data management mode of computer technology, such as p2p transmission, distributed data storage, encryption algorithm, and consensus mechanism, the blockchain technology is integrated into the MEC environment architecture. Also, a blockchain-enabled distributed video content caching and transcoding framework is proposed. In this framework, edge nodes perform video content caching and transcoding and the CDN tips maintain a licensed blockchain to ensure the integrity and accuracy of cached video data, leading to design the best video caching and transcoding scheme.

3. System Model Design

For convenience, the major notations used in this article are summarized in Table 1.

3.1. BI-CMEC System Model Design. To satisfy the requirements for video distribution across the entire network and maximize the role of edge computing nodes, it is necessary to perfectly integrate edge nodes with the existing wireless network environment and video transmission technology, fully cooperating with cloud center and user terminals

[44–50]. At the same time, the blockchain technology can be integrated into the mobile edge computing environment. The blockchain can be used as an auxiliary framework to manage the provision of mobile edge computing resources and turn the supply of edge computing resources into a blockchain application.

In Figure 1, this paper intends to present a blockchain-enabled CMEC-based video transmission model. The first important aspect is the Clustered Mobile Edge Computing (CMEC) system. In this CMEC system, mobile edge computing is seamlessly connected with CDN tips. Also, the CDN tips and some mobile edge computing nodes make the mobile edge area. Based on the CMEC model, the storage, computing, and communication capabilities in one cluster can be collaboratively used to optimize wireless video streaming transmission quality through the whole network.

In this system, the CDN tip is a central server directly connected to each edge cluster. It can provide the original cached video resources for each edge cluster and the computing power support required for the implementation of deep reinforcement learning algorithms and is also responsible for maintaining the permission blockchain to ensure the transparency, security, privacy, and integrity of cached video data and user information. Also, the proposed blockchain-enabled CMEC-based video transmission model in this paper will seamlessly enable the blockchain scheme into our Clustered MEC network, combining with the popular CDN video transmission system by connecting edge clusters with CDN tips.

Furthermore, the second important aspect is the blockchain-enabled intelligent video caching and transcoding framework for CMEC-based video transmission system. In this framework, mobile edge nodes perform joint video caching and transcoding by using the deep reinforcement learning method in one mode (independent or federated) and the CDN tips will maintain a licensed blockchain to ensure the transparency, security, privacy, and integrity of cached video data and user information. The mobile edge nodes in the cluster, which can participate in the task of intelligent video caching and transcoding, will depend on the security model of the blockchain.

The security model of the blockchain includes four levels of the encryption guarantee layer, the consensus guarantee layer, the economic guarantee layer, and the social security layer. Encryption guarantee is used to ensure that only safe and legal edge nodes can participate in the task of intelligent video caching and transcoding services. The consensus mechanism layer is to complete the verification and confirmation of the transaction in a short time through the voting of trusted edge nodes to ensure the accuracy of the information. Economic security can reward good behavior (with block rewards and fees) and punish bad behavior (by cutting margin or withholding future rewards). Also, social security is the last guarantee. If the consensus attack exceeds the stage of economic security, the society can still reject it by manually lifting the control of the miner.

So, the blockchain designed in this article has four functions as follows:

TABLE 1: Summary of major notations.

Notation	Description
t	The time stage
$M(t)$	The serial number of MEC servers in a cluster
$U(t)$	The decision of video cache updating
z_v	The probability of the requests of video v
$C_b(t)$	The corresponding bandwidth cost
$P(n, t)$	The unit bandwidth price
$W(n, t)$	The amount of bandwidth usage in the MEC server n
$B(i, t)$	The bitrate assigned to user i at time slot t
$O(t)$	The video transcoding cost
$L(t)$	The buffer occupancy rate
$R(t)$	The video rebuffering time of playback buffer
$q(t)$	The video quality rate
$CRH(t)$	The video cache hit rate
$(pk_{i,j}, sk_{i,j})$	The public/private key pairs of user mobile device i connecting to edge node j
(pk_j, sk_j)	The public/private key pairs of edge node j
$ID(V_{i,j})$	The ID of video content $V_{i,j}$
$H(ID(V_{i,j}))$	The generated hash value of $ID(V_{i,j})$

- (a) *Transaction Transparency*. Any party in this network system (including users, mobile edge nodes, and CDN tips) can choose to download and access this blockchain to obtain information about cached and transcoded video transactions.
- (b) *System Security*. The blockchain uses asymmetric encryption technology, so that the security of the blockchain can be achieved by using private and public keys. Blockchain can provide identity verification function and access control to protect CMEC system.
- (c) *User Information Privacy*. The user privacy in the blockchain is achieved by public key anonymity. Immutability and consensus ensure the privacy of the database stored on the blockchain
- (d) *Video Data Integrity*. As users and edge nodes reach a consensus, edge nodes and users will add blocks of newly completed transactions (smart contracts) broadcasted by honest entities to the blockchain. The consensus mechanism not only ensures the integrity of the video data but also makes the content of the video data transparent in the blockchain network formed by all nodes, that is, unmodifiable means to monitor the integrity and authenticity of data.

3.2. Video Caching Model. In the CMEC system, to reduce the delay time of video streaming services, the more popular video content is proactively cached in the network edge node. In our CMEC system, the video content is actively cached in any edge MEC server in one edge cluster. Videos with high video popularity are generally cached at edge MEC servers, while videos with low video popularity are generally cached at CDN tips. In a cluster, the specific location of the video cached with high video popularity needs to be intelligently selected according to users' needs. In CMEC system, the video cache updating, which means the caching action at each time stage t , is denoted as $\text{Cache}(M(t), U(t))$.

According to this caching model, the videos are needed to be proactively cached according to the video popularity, which is the user's preference for the video. Therefore, the popularity of the video content will be requested to be obtained as the basic data information. The important key to solve the video caching problem is videos' popularity distribution. During the caching process, the video content cached by the MEC server also needs to be updated continuously. In our caching model, we update the caching video by using the first-in-first-out (FIFO) method, selecting the more popular videos at the mean time. Then, the probability of video v is defined as

$$Z_v = \frac{v^{-\alpha}}{\sum_{v=1}^V v^{-\alpha}}, \quad (1)$$

where $\alpha > 0$ is the parameter in Zipf distribution, indicating the skewness degrees [51].

In addition, the corresponding bandwidth cost will be generated during the caching process, which is represented by the symbol $C_b(t)$. Bandwidth cost is one of the main costs that need to be considered. In the CMEC system, we suppose that the price of bandwidth remains constant within a time stage. Then, the bandwidth cost $C_b(t)$ [52] of MEC servers in one cluster can be defined as follows:

$$C_b(t) = \sum_{n=1}^M P(n, t) \cdot W(n, t), \quad (2)$$

where M is the number of MEC servers in one cluster at time stage t . Also, the bandwidth cost in MEC server n can be computed by the following formula:

$$W(n, t) = \sum_{i \in U^t} B_u(i, t) \cdot I^t(i, n), \quad n \in \{0, \dots, M-1\}, \quad (3)$$

where $q(t)$ is the user group in one cluster at time stage $q(t) = \beta \log(B(t))$. The symbol β is an indicator, representing if user $CRH(t)$ needs be connected to the MEC server T at time stage t .

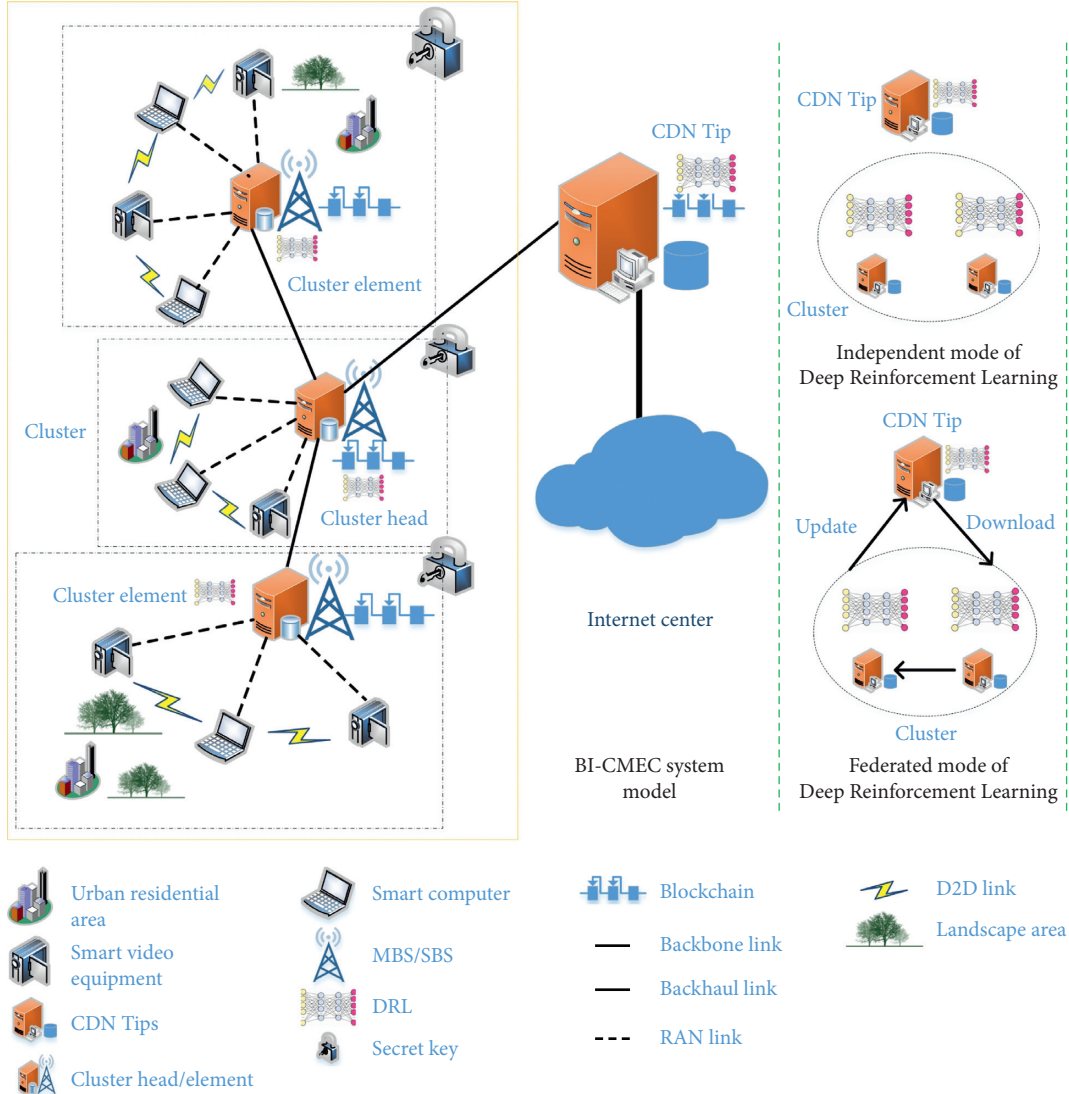


FIGURE 1: Blockchain-enabled intelligent video caching and transcoding framework for Clustered Mobile Edge Computing system.

3.3. Video Transcoding Model. To reduce the delay time of video services and improve the QoE of video services, proactive popular video caching has been carried out on the network edge server. However, in real life, different users have different network conditions, which leads to different users' needs for the same video. That is, users with good network conditions need high-definition video sources, while users with poor network conditions only need video sources with general definition. This requires the network edge server to cache different versions of the video source for the same video, but this requires a huge caching space to satisfy.

Therefore, it is considered that the functions of caching and transcoding can be realized at the CMEC network edge server at the same time. Thus, the CMEC network edge server can perform video transcoding in real time according to different user needs and actual conditions to satisfy the needs of different users, improving the efficiency and QoE of video services.

In our CMEC system, let $B_u(i, t) \in \{B_1, B_2, \dots, B_{\max}\}$ be the set of all video layers in video transcoding service. Also, the symbol B_{\max} represents the original video level cached at

the MEC server. So, the video transcoding action is video transcoding layer decision and which MEC server in the cluster will carry out this task of video transcoding at each time stage t .

Because the network edge server will implement video transcoding in real time according to different user needs and actual conditions, the video services also need to consider the transcoding cost. Generally, the target video bit-rate, the input video bit-rate, the number of CPU cores, and the video length will closely affect the transcoding cost. These need to be considered in the video pricing model. Then, the video transcoding cost can be defined as

$$O(t) = \sigma * (L_{\max} - l) * T_v * N_{\text{cpu}}, \quad l \in \{L_1, L_2, \dots, L_{\max}\}, \quad (4)$$

where the symbols l , N_{cpu} , T_v , and σ represent level of input video data, the number of CPU cores required for transcoding task, the length of video, and an adjustable parameter, respectively.

3.4. Rebuffer Model. The video playback buffer is usually set on the user's smart device to ensure continuous playback of the video, in which the video block will be downloaded to the buffer [53]. Let $W(t)$ represent the wireless transmission rate of smart device users. Also, the symbol $B(t)$ denotes the bitrate of the chunk of the video data. So, the buffer occupancy rate $L(t)$ will be obtained as

$$L(t) = \frac{\text{buffer occupancy}}{\text{buffer size}}. \quad (5)$$

Furthermore, in the rebuffer model, the buffered video time and rebuffering time could be usually introduced [54]. The video rebuffering time of playback buffer is denoted as $R(t)$:

$$R(t) = \max(d(t) - T(t), 0), \quad (6)$$

where $T(t)$ denotes the buffered video time at playback buffer at the beginning of time stage t . Also, the total downloading time of one chunk during time stage t is denoted by $d(t)$.

3.5. Video Quality-Rate and Cache Hit Rate Model. Under normal circumstances, for video quality evaluation, the video quality of a rate-encoded video can be approximated by the following logarithmic function [55]:

$$q(t) = \beta \log(B(t)), \quad (7)$$

where the value β will be obtained from the video encoder when encoding in the video source. Furthermore, the high-definition video generally has a higher bit rate, while the standard-definition video generally has a lower bit rate.

In addition, the quality of video streaming service can generally be evaluated and analyzed using the cache hit rate. The cache hit rate of T requests in the time period [51] can be obtained as

$$CRH(t) = \frac{\sum_{i=1}^T 1(H_i)}{T}, \quad (8)$$

where $1(H_i)$ is an indicator function.

4. Blockchain Empowerment Mode

4.1. Integration of Blockchain and Video Streaming Service Network. Blockchain is a layered architecture, which includes data layer, network layer, consensus layer, incentive layer, contract layer, and application layer. Also, it is a decentralized system composed of P2P networks. In our CMEC system, the CDN tip and some mobile edge computing nodes make the mobile edge area. Because of the large difference in capabilities among mobile edge nodes, a clustered mobile edge computing model is introduced to cluster edge nodes at the network local area. Then, there are also some mobile edge nodes in one cluster, and one cluster head is selected based on the storage, computing, and communication capabilities of the network edge node. The users can also be divided into some different user groups in one mobile edge area, according to user preferences for

videos. Generally speaking, users in the same group often have the same type of edge nodes to provide video streaming services.

As shown in Figure 2, the detail of blockchain-enabled process for intelligent video streaming service has been given as follows:

- (a) Requesting and making decision: when there is a request, the requested edge node uses the deep reinforcement learning method to make a video caching and transcoding decision based on the user's request content and the actual network environment and also returns the decision result to the user
- (b) Create smart contract: according to the decision, the video caching and transcoding smart contracts are created based on user requests and task requests in the edge cluster
- (c) Execute smart contract: the smart contract will be executed to realize the edge nodes in the edge cluster to cache the video, complete the video transcoding task, and provide the transcoded requested video to the user
- (d) Record and release the smart contract: finally, put the smart contract transaction data on the blockchain and release the smart contract

In the CMEC system, video streaming services can be implemented by P2P connection between edge nodes and user equipment. All the edge nodes and user equipment are made to be consensus nodes. Only the trusted consensus nodes are able to access CMEC video streaming service system, in which the consensus nodes should have passed authorization in blockchain. In the CMEC system, we need to consider where to cache the requested video and choose which version of the requested video should be transcoded for the users. However, in real life, different users have different network conditions, which leads to different users' needs for the same video.

Therefore, selecting the appropriate video resolution according to the user's real-time network conditions is a relatively complex decision-making problem in the high-dimensional state space.

To solve this problem, we use a framework based on deep reinforcement learning to automatically acquire intracluster collaborative caching and transcoding decisions. These decisions are executed on real time based on user demand predictions, video data popularity, and the capabilities of the MEC server. Based on the blockchain technology, we use smart contracts to implement edge collaborative caching and transcoding applications. Then, the packaging node uploads the key information of the edge collaborative cache and transcoding application to the blockchain to save the certificate and ensure efficiency and accuracy. Through the smart contract, the video streaming service transaction between the edge node and the user is realized. The user pays a certain token, and the edge node that provides the service can receive the token reward.

By blockchain-enabled framework for CMEC-based video transmission system, the blockchain technology is introduced into the CMEC system to ensure the transactions'

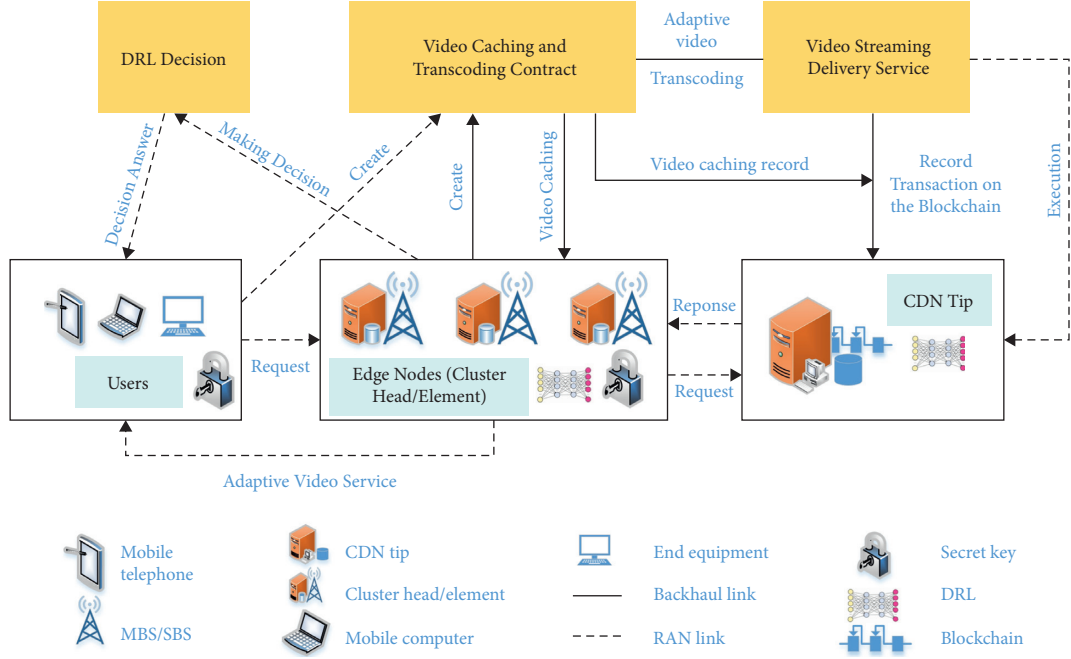


FIGURE 2: Blockchain-enabled process for intelligent video streaming service.

transparency, system security, user information privacy, and integrity of the video data. Also, the intelligent video streaming service has been provided, which has a series of smart contracts to acquire self-organization at video caching and transcoding services, especially without a centralized controller, only based on real-time DRL decision results.

In this system model, deep reinforcement learning is generally performed at the edge nodes in the cluster. Also, in blockchain-enabled CMEC-based video transmission system, the edge nodes are generally macro base station or small base station which is equipped with servers. Their computing power is strong enough to quickly implement learning algorithms and give decisions. The delay time of this process is much shorter, which can be ignored in comparison. Furthermore, the real-time computing power required for video caching and transcoding is not high. Even when the required computing power is higher, the CDN tip directly connected to the edge cluster can help to complete the task and quickly return the calculation result. Therefore, we need not consider the latency of the process of deep reinforcement learning in the experiments.

4.2. Joint Video Caching and Transcoding Decision Smart Contract Design. In the blockchain-enabled video streaming service, we use tokens to realize the video streaming service transaction [11]. Since the video streaming service interaction between network edge nodes and smart mobile device users is not supervised by other parties, malicious users will deliberately refuse to pay the caching and transcoding service fees of edge nodes. Based on this, first, we plan to use the blockchain technology to supervise the video caching and transcoding transactions between network edge nodes and smart mobile device users in a distributed mode. In this paper, we use blockchain to realize collaborative caching and

transcoding transactions among network edge nodes and smart mobile device users. The joint caching and transcoding transaction between network edge nodes and smart device users is implemented in blockchain by a smart contract. The smart contract is specially designed to manage the transaction process of the joint caching and transcoding service, which records the joint caching and transcoding transaction and payment information on a continuous blockchain. In particular, the smart contract can check the integrity of results returned from the network edge node to achieve adaptive video transmission optimization and make security protection of video data in the blockchain-enabled CMEC-based environment.

The detail of the joint video caching and transcoding decision smart contract construct is shown in Algorithm 1.

- (a) Initialization: the initialization is the setup function of the joint video caching and transcoding service, where the user's smart mobile device i interacts with the network edge node j . The initialization formulates smart contracts for the joint video caching and transcoding service. The user smart mobile device i and network edge node j both generate public and private key pairs, which are represented by $(pk_{i,j}, sk_{i,j})$ and (pk_j, sk_j) , respectively. Every smart contract contains a set of variables, including cached video location $M_{i,j}$, cache update decision $U_{i,j}$, cached video content size $S_{i,j}$, service user video version $B_u(i, t)$, network edge node transcoding price p_j , deployment time $dTime$, timestamp $tStamp$, and contract service period time $cTime$. Finally, in this function, the user smart mobile device i and the network edge node j will both sign the contract by using their own private keys $sign(ID_{i,j})$ and $sign(ID_j)$.

- (b) Create: after the user smart mobile device i and the network edge node j reach an agreement, they will deploy a new smart contract by using the creating function on the blockchain. The output of the creating function is the contract address on the blockchain, which is public to all network edge nodes and smart device users. To ensure the execution of the smart contract and prevent malicious behavior, both the user smart mobile device i and the network edge node j must submit certain deposits from their own accounts to the smart contract. The paid deposits can be represented by $\text{deposit}_{i,j}$ and deposit_j , respectively.
- (c) Transaction execution: if a specific smart contract has been set on the blockchain, the transaction function is executed. When the user smart mobile device i requests a video streaming from the network edge node j , based on the user network environment and the load of the nodes at the cluster in CMEC system, using the deep reinforcement learning algorithm, the user mobile device $Q(s, a; \theta)$ should generate a cooperative caching and transcoding token with the help of the edge device; its value is $\delta_{i,j} = (\text{ID}(V_{i,j}), M_{i,j}, U_{i,j}, B_u(i, t), h_{i,j})$, where $\text{ID}(V_{i,j})$ is the ID of video content $V_{i,j}$ and $h_{i,j}$ is $H(\text{ID}(V_{i,j}))$, the generated hash value, where the symbol H is the public hash function. After that, the user smart mobile device signs $\delta_{i,j}$ by $\text{sign}(\delta_{i,j})$. The user smart mobile device sends the signature $\text{sign}(\delta_{i,j})$ and $\delta_{i,j}$ to the network edge node j . Once the network edge node j receives $\delta_{i,j}$, it will firstly use the public key $pk_{i,j}$ of the user smart mobile device to verify $\delta_{i,j}$. Simultaneously, verify $h_{i,j}$ and implement collaborative caching and transcoding in $\delta_{i,j}$. After completing the above process, network edge node j generates the following joint video caching and transcoding transaction:

$$\text{Transaction} \longrightarrow (h_{i,j}, \text{sign}(\delta_{i,j}), M_{i,j}, U_{i,j}, B_u(i, t), p_j). \quad (9)$$

- (d) Recording: after the transaction is completed, the network edge node j in cluster sends the transaction to the selected smart contract. The network edge node of cluster heads and user group leaders in the network will use the DPOs consensus protocol to record transactions on the blockchain.
- (e) Penalty and settlement: at this stage, because the smart contract can monitor the video content transmission service between the user's smart mobile device and the network edge node, if any of them does not abide by the signed agreement, the penalty function will be called to implement the penalty. Finally, when the smart contract is completed and reaches the service period, financial settlement will be performed and all assets owned by the smart contract will be released.

5. Problem Formulation and Intelligent Video Caching and Transcoding Scheme

To use deep reinforcement learning algorithm for network resource optimization at the BI-CMEC system, we used the DQN method to address the joint video caching and transcoding optimization problem.

5.1. Problem Formulation. In this paper, the objective function of the joint video caching and transcoding issue is to maximize the expected average reward. Therefore, we model the dynamic optimization problem as a Markov Decision Process, which is

$$\begin{aligned} \max_{M(t), U(t), B_u(i, t)} J(t) &= E \left[\sum_{t=0}^{T-1} \gamma^t r(t) \right], \\ \text{s.t. C1: } M(t) &\in \{0, 1, \dots, M\}, \forall t, \\ \text{C2: } U(t) &\in \{0, 1\}, \forall t, \\ \text{C3: } B_u(i, t) &\in \{B_1, B_2, \dots, B_{\max}\}, \forall t, \end{aligned} \quad (10)$$

where $\gamma \in [0, 1]$ is a discount factor and $r(t)$ is the reward at the time stage t in this optimization problem.

In general, it is difficult to solve optimization problems with a large number of states in the state space. However, for large-scale optimization problems that do not require any prior knowledge of state transition probability, the DRL algorithm has been proven to be a very effective mathematical tool.

5.2. DQN for Intelligent Video Caching and Transcoding Decisions. The basic idea of many reinforcement learning algorithms is to gradually estimate the Q value function by using the Bellman equation as an iterative update, so that

this value iterative algorithm converges to the optimal Q value function. Since the advent of the deep Q network [56–58] in 2013, many scholars have chosen to use the DQN algorithm as an algorithm and method to solve practical application problems. The value iteration algorithm to solve the optimal Q value function is as follows:

$$Q_{i+1}(s, a) = \mathbb{E}[r + \gamma \max_{a'} Q_i(s', a') | s, a]. \quad (11)$$

In DQN, Mnih et al. refer to a neural network function approximator with weights θ as a Q-network, which can be trained by minimizing a sequence of loss functions $L_i(\theta_i)$ that changes at each iteration i .

$$L_i(\theta_i) = \mathbb{E}_{s,a \sim \rho(\cdot)} \left[(y_i - Q(s, a; \theta_i))^2 \right], \quad (12)$$

$$y_i = r + \gamma \max_{a'} Q(s', a'; \theta_{i-1}) | s, a,$$

where y_i is the target for iteration i and $\rho(\cdot)$ is a probability distribution over sequences and actions which we refer to as the behavior distribution.

In deep reinforcement learning of DQN method, there are three basic elements, which are the action, state, and reward of the optimization issue. In our joint video caching and transcoding optimization issue, they can be obtained as follows:

$$\begin{aligned} S(t) &= \{C_b(t), L(t), R(t), q(t)\}, \\ A(t) &= \{M(t), U(t), B_u(i, t)\}, \\ r(t) &= \omega_1 CRH_{sl}(t) + \lambda q(t) - \omega_2 \|q(t) - q(t-1)\| \\ &\quad - \omega_3 R(t) - \omega_4 C_b(t) - \omega_5 O(t), \end{aligned} \quad (13)$$

where in the state $S(t)$, $C_b(t)$ is the current bandwidth cost, $L(t)$ is the current buffer occupancy rate, $R(t)$ is the current playback buffer, and $q(t)$ is the current video quality downloaded during time stage t . Also, the action is selected from the action set $A(t)$. Lastly, at the reward, the weighted sum of the short- and long-term cache hit rate $CRH_{sl}(t)$ for each step is obtained as

$$CRH_{sl}(t) = CRH_s(t) + \mu * CRH_l(t), \quad (14)$$

where μ is the weight to balance the short- and long-term cache hit rate.

In the reward, it consists of video quality, video quality variation, video playback rebuffering time, and two penalty. Furthermore, the user perceived QoE in video streaming service is directly depended by the total cache hit rate, video quality variation, video quality, and video playback rebuffering time. Symbols $\omega_1, \lambda_1, \omega_2, \omega_3, \omega_4$, and ω_5 are the weighting parameters in the formula.

5.3. Intelligent Video Caching and Transcoding Scheme. Our proposed intelligent video caching and transcoding scheme has a series of smart contracts which can acquire self-organization at video caching and transcoding services, especially without a centralized controller. The proposed intelligent video caching and transcoding scheme has two sections, namely, DQN-Based Video Caching and Transcoding Algorithm (DQN-VCT) (section 1) and Implementation of Video Caching and Transcoding Decision Smart Contract (section 2).

In section 1, there are two major factors that support DQN and make it extremely powerful. The two major factors are experience replay and fixed Q-targets. Through these two factors, the correlation between the learning samples is removed, and the learning efficiency of DQN is getting higher and higher.

In DQN-Based Video Caching and Transcoding Algorithm (DQN-VCT), the inputs of the deep neural network are the video service system states $S(t)$, which are listed in equation (13), and the outputs of the network are the Q value

function, $Q(s, a; \theta)$ for each action are listed in equation (11). Based the method in our previous article [59], we illustrate the details of the DQN-based video caching and transcoding algorithm in section 1 in Algorithm 2.

Then, based on the decision of video caching and transcoding in section 1 by the algorithm, the selected edge node in cluster will execute automatically the implementation of video caching and transcoding decision smart contract. The smart contract will be strictly executed according to Algorithm 1.

6. Simulation and Analysis

This section contains two parts. First, the experiment settings were illustrated. Then, the experimental simulations were carried out to prove the performance of the proposed scheme.

6.1. Experimental Settings

6.1.1. Data Generation. In the experiments, the smart device user data of requests will be generated randomly. The video data of smart device users' requests were generated under the Zipf function distribution. Different numbers of requests in one episode have been adopted as the testing data, such as 50, 70, and 100. The video data in smart device users' different numbers of requests were obtained by unchanged popularity distribution, in which the Zipf function parameter is set as 1.3.

6.1.2. Parameters Setting. In the experiments, we deploy 7 MEC network nodes in one cluster, which will serve 30 smart device users in this region and also provide about 50 videos for smart device users' requests. There are four video layers of the video in the experiment, with the original layer at the MEC node in the cluster as B_{\max} . The video transcoding from B_{\max} to B_1, B_2 , and B_3 will need, respectively, 2, 4, and 6 CPU cycles. Then, we set the parameter in the experiments as given in Table 2.

6.1.3. Deep Neural Network Setting for DQN. In the experiments, a fully connected neural network was adopted, which consists of 2 hidden layers, 256 and 512 in size, respectively. The loss function we used was the mean square error function. The naive ϵ -greedy strategy was adopted for exploration, in which the probability of randomly choosing an action during the training stage was ϵ . The degree of exploration continues to shrink when the learning progresses. The size of experience replay in DQN and the learning rate were adopted as 2000 and 0.01. Also, the number 0.90 was chosen as the attenuation parameter which is used to update the target Q network. Then, the batch size in stochastic batch gradient descent was 32. Finally, the experiment simulations were carried out by using Python.

6.1.4. Environment Setting for Blockchain. To assist the simulation experiment, in the simulation implementation of

- (1) **Initialization:**
- (2) Initialize the input data $ID_{i,j}, ID_j, M_{i,j}, U_{i,j}, S_{i,j}, B_u(i, t), p_j$
- (3) Initialize state $\{(pk_{i,j}, sk_{i,j}), (pk_j, sk_j), dTime, tStamp, cTime\}$
- (4) sign $(ID_{i,j})$ and sign (ID_j) on the selected smart contract
- (5) **Creat:**
- (6) Output the smart contract address
- (7) Input: $deposit_{i,j}, deposit_j$
- (8) Verify: $deposit_{i,j} \geq p_j, deposit_j$
- (9) **Transaction Execution:**
- (10) Verify the state: $t > dTime$
- (11) Edge cluster asks the video for caching
- (12) The selected edge node in edge cluster implements the transcoding task
- (13) Edge cluster sends the appreciate video to users
- (14) Edge cluster broadcasts collaborative caching and transcoding transaction
- (15) Transaction as:
- (16) Transaction $\longrightarrow (h_{i,j}, \text{sign}(\delta_{i,j}), M_{i,j}, U_{i,j}, B_u(i, t), p_j)$
- (17) **Recording:**
- (18) Edge node in cluster sends the transaction to the selected smart contract
- (19) Edge cluster and users in the network use the DPoS consensus protocol to record transactions on the blockchain
- (20) **Penalty and Settlement:**
- (21) Verify the state: $t > cTime$
- (22) Penalty execution: $penalty_{i,j}, penalty_j$
- (23) Settlement:
- (24) $(ID_{i,j}, deposit_{i,j} - p_j + penalty_{i,j}), (ID_j, deposit_j + p_j + penalty_j)$

ALGORITHM 1: Joint video caching and transcoding decision smart contract algorithm.

the blockchain, we use Ganache to simulate the operating environment of Ethereum and deploy smart contracts through Truffle. When using Truffle to deploy a smart contract, the network address and network number of the current simulation environment will be used so that Truffle can deploy the smart contract to Ganache's Ethereum test environment through this interface.

6.2. Experimental Simulation Results. In this section, we perform intelligent video caching and transcoding by deep reinforcement learning method in the independent mode. Also, we compare the proposed BI-CMEC scheme (called BI-CMEC method) with the CMEC scheme without blockchain technology (called CMEC method). Because of the characteristics of DRL, for the proposed algorithm and compared method, all the reported results would be acquired from the average of 20 algorithm executions.

Figure 3 shows the convergence performance of DQN-VCT algorithm, which is with the set of full weight at different learning rates. From Figure 3, we can see that the performance of learning rate 0.01 is the best among the three different learning rates. It is better than the performance with learning rate 0.1 and 0.001. Because a large update step will lead the average reward converging to a local optimal solution, convergence performance in learning rate 0.1 becomes worse. Generally, the appropriate learning rate always depends on the real-time state of the environment at the current optimization step.

Figure 4 shows the comparison of the QoE value on video streaming service performance with and without blockchain empowerment. It can be seen from Figure 4 that

the QoE value of the BI-CMEC method empowered by the blockchain is slightly worse than that of the CMEC method. This is because the blockchain empowerment introduces a consensus mechanism, which causes a certain time delay, reducing the QoE value of video streaming. Comparing to the CMEC method, blockchain empowerment brings security and privacy protection to video streaming service systems based on edge computing in the BI-CMEC method. Based on blockchain-enabled framework for CMEC-based video transmission system, the blockchain technology is introduced into CMEC system to ensure the transactions' transparency, system security, user information privacy, and integrity of the video data.

It can be seen from Figure 5 that the bandwidth cost performance between the BI-CMEC method and CMEC method is much similar. At the beginning, the service cost of the two methods is relatively high, but as the learning process continues to advance, the service cost slowly decreases. Although the BI-CMEC method enabled by blockchain has a higher service cost than the CMEC method in the later stage, overall, the service cost of the BI-CMEC method enabled by the blockchain is very similar to that of the CMEC method. This is because there is no other cost load that is introduced in the BI-CMEC method, except time delay comparing to the CMEC method.

In order to better analyze the experimental effect, we give a comparison of the bandwidth cost and the average QoE when the blockchain is placed in different locations. From Figure 6, we can see that when the blockchain is set in the CDN tip, the bandwidth cost is slightly higher than that of the other two locations, namely, cluster head and cluster element. This is because when many clusters' blockchains are

(1) Section 1: DQN-Based Video Caching and Transcoding Algorithm (DQN-VCT)
(2) Initialization:
(3) Initialize the replay memory D to capacity N
(4) Initialize the Q network and the target Q network with random weights
(5) Initialize MEC network service matrix V of requests
(6) for episode = 1, M do
(7) Generate the smart device users' requests data
(8) Observe initial state s_1 as illustrated in equation (13)
(9) for $t = 1, T$ do
(10) Give a random probability $\zeta \in [0, 1]$
(11) Choose action $A(t)$ which listed in equation (13) as $A(t) = \begin{cases} a^*(t) = \arg \max_a Q(s, a; \theta), & \zeta > \varepsilon, \\ a(t) \neq a^*(t), \text{ randomly select } a(t), & \text{others.} \end{cases}$
(12) Observe the reward $r(t)$, state $s(t+1)$
(13) Store the transition $(s(t), A(t), r(t), s(t+1))$ into Buffer pool D
(14) Update MEC network service matrix V of requests
(15) Sample random minibatch of transitions $(s(t), A(t), r(t), s(t+1))$ from Buffer pool D
(16) Set $y_j = \begin{cases} r_j, & \text{for terminal } s', \\ r_j + \gamma \max_{a'} Q(s', a'; \theta_{i-1}) s, a, & \text{o non-terminal } s'. \end{cases}$
(17) Implement a gradient descent step according to equation (12)
(18) Update the parameters within the Q network
(19) Reset the parameters within the target Q network every G time stages
(20) end for
(21) end for
(22) Section 2: Implementation of Video Caching and Transcoding Decision Smart Contract
(23) Create SC:
(24) Trigger the smart contract according to the user's request based on the action $A(t)$ from Section 1
(25) Execute the SC: carry out the smart contract for managing the transaction process of the intelligent caching and transcoding service
(26) Record the SC: record transactions on the blockchain
(27) Release the SC: the smart contract is released.

ALGORITHM 2: Intelligent video caching and transcoding scheme.

TABLE 2: Summary of major parameter values.

Parameter	Value	Parameter	Value
D	10 s	ω_3	0.1
β	6.5	ω_4	0.1
α	1.3	ω_5	0.1
μ	0.6	B_{\max}	10 Mbps
σ	1.2	B_1	1 Mbps
ω_1	1.2	B_2	2 Mbps
λ	1.2	B_3	4 Mbps
ω_2	0.9	CPU cores	{2, 4, 6, 8}

set in the CDN tip compared to the cluster head and cluster element, it will cause more bandwidth consumption. Then, it can be easily seen in Figure 7 that as the learning process continues, the position of the blockchain has less impact on the video QoE and tends to be similar.

Then, we analyzed the experimental results when there are different numbers of user requests in an episode. In Figure 8, overall, in order to pursue higher video quality at the beginning, the bandwidth cost is higher. When the number of smart device users' requests in a time slot is 100, the bandwidth cost generated is the largest. With continuous learning, the bandwidth cost is slowly reduced and the balance between bandwidth cost and video QoE is desired. Similarly, it can be directly seen in Figure 9 that different numbers of user requests in an episode have almost no effect on the QoE value. Based on the above experimental results,

we can find that the BI-CMEC method we proposed has better robustness to the Internet environment.

Furthermore, the framework and algorithm of this article can withstand the attacks that blockchain technology can withstand, such as encryption cracking, consensus mechanism challenges, 51% attacks, and N@S (nothing at stock) attacks. The discussion of these attacks is well documented in the blockchain theory and technology related literature. Also, the topic of this article is blockchain-enabled intelligent video caching and transcoding in clustered MEC networks, which integrates blockchain technology and applies it into video streaming services based on edge computing to ensure user information privacy and video data security through the distributed storage structure of blockchain. The core point in our paper is how to effectively integrate blockchain technology into intelligent

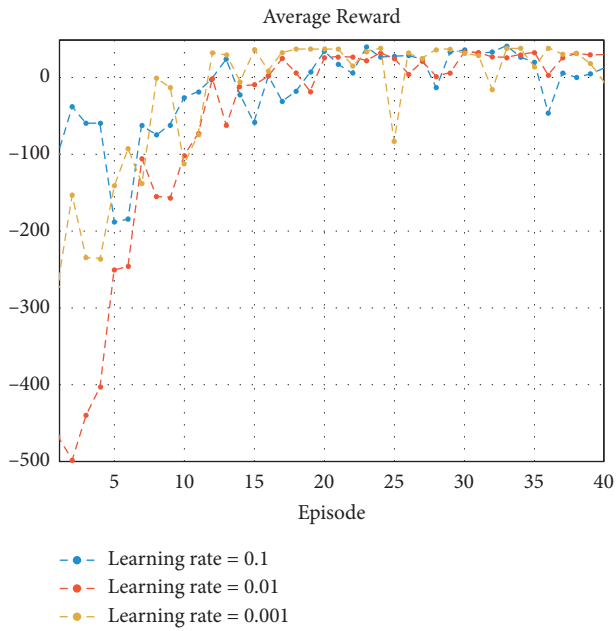


FIGURE 3: The convergence performance of DQN-based joint video caching and transcoding algorithm with the different learning rates.

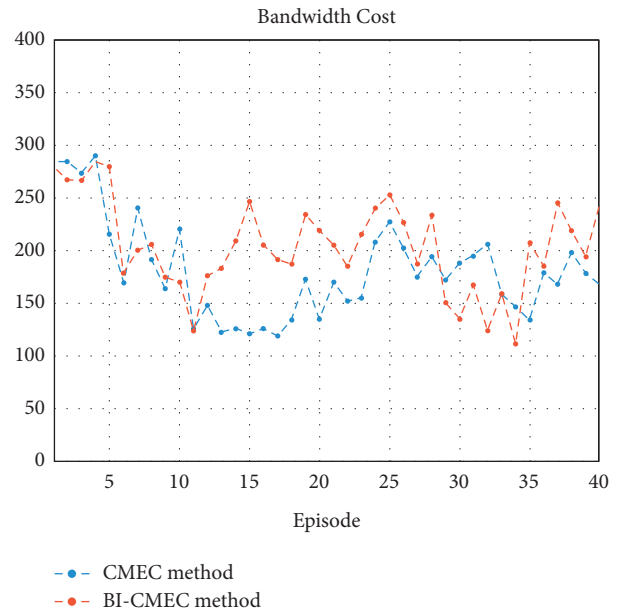


FIGURE 5: The bandwidth cost performance between the BI-CMEC method and CMEC method with blockchain at CDN tip and 50 requests in an episode.

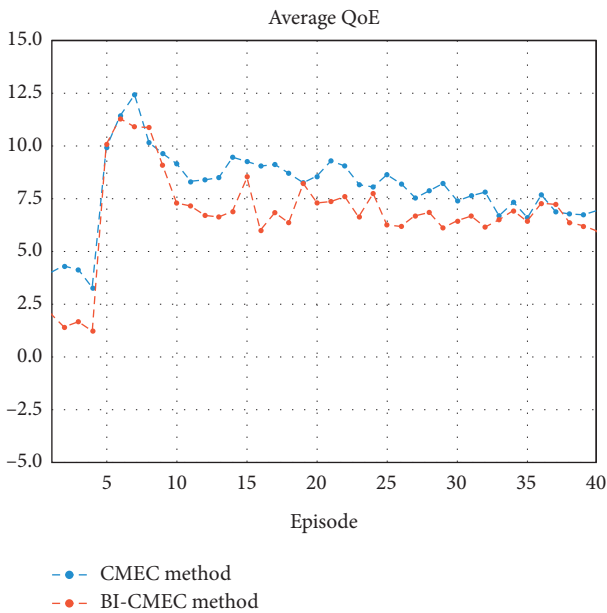


FIGURE 4: The QoE performance between the BI-CMEC method and CMEC method with blockchain at CDN tip and 50 requests in an episode.

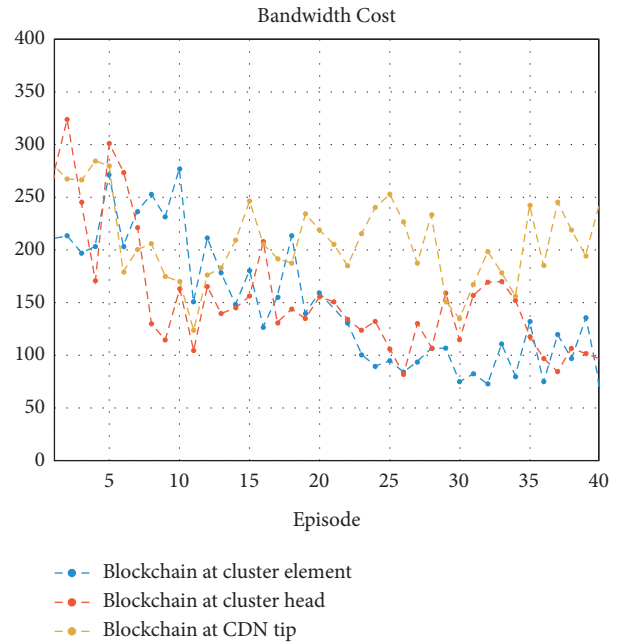


FIGURE 6: The bandwidth cost performance in the BI-CMEC method with blockchain at different positions.

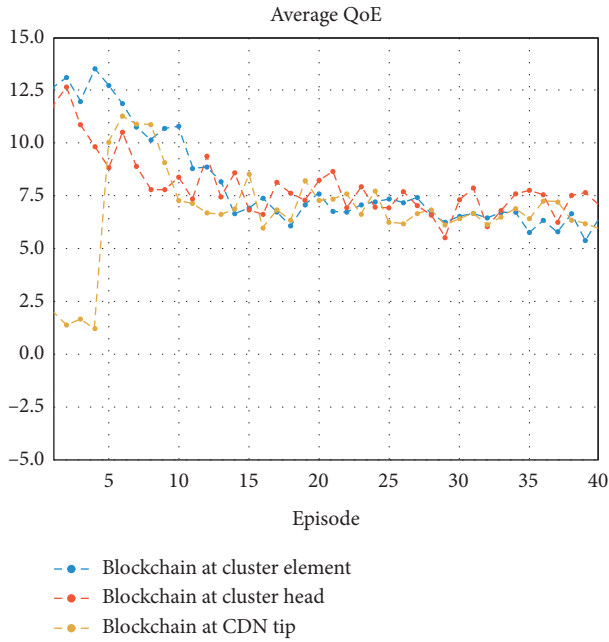


FIGURE 7: The QoE performance in the BI-CMEC method with blockchain at different positions.

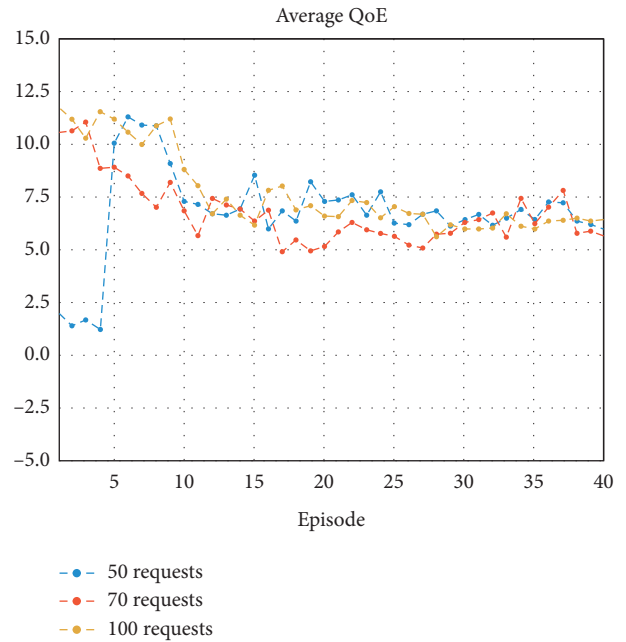


FIGURE 9: The QoE performance of the BI-CMEC method in different requests' numbers at an episode.

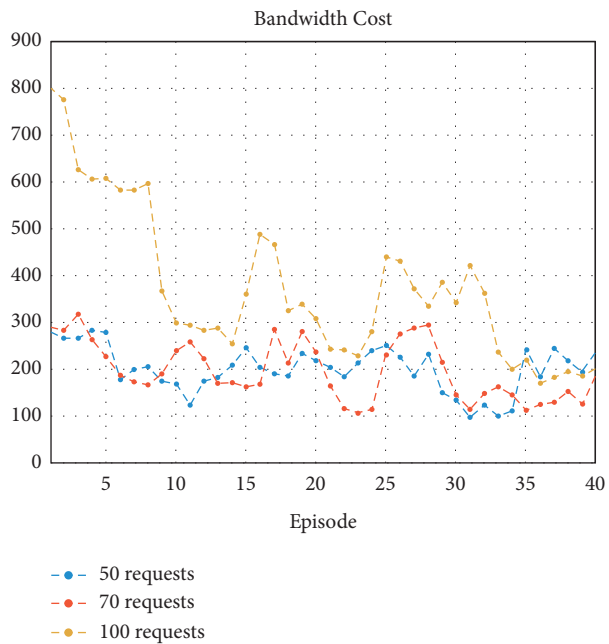


FIGURE 8: The bandwidth cost performance of the BI-CMEC method in different requests' numbers at an episode.

video services in the clustered edge computing environment, on the basis of improving security performance and ensuring the high quality of intelligent video services. Therefore, the discussion of these attacks will not be repeated here.

7. Conclusions

In this paper, we firstly proposed a blockchain-enabled CMEC-based video transmission system model (BI-CMEC) that could ensure the transactions' transparency, system security, user information privacy, and integrity of the video data, enhance the ability of servers in actively caching popular video content in the CMEC system, and realize transcoding function at network edge nodes. In addition, we proposed an intelligent video caching and transcoding scheme. A smart contract is specially designed which can acquire self-organization at video caching and transcoding services, especially without a centralized controller. Furthermore, we adopted a DQN-based framework to automatically obtain the intra-cluster joint video caching and transcoding decisions. Finally, the experimental results were presented to validate the effectiveness of the proposed method.

Based on the model of BI-CMEC system, this paper mainly focuses on ensuring video data security and user privacy protection and also encouraging the collaboration among MEC network nodes in one cluster. In this model, the DPoS consensus protocol was used in video transmission application scenarios. In the future work, we will perform intelligent video caching and transcoding by using the deep reinforcement learning method in the federated mode, and more efficient consensus mechanism algorithms need to be proposed to meet the special needs of video transmission application scenarios.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that there are no conflicts of interest.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (Nos. 61961021 and 52165015), Science and Technology Project of Jiangxi Education Department (Nos. GJJ180251 and GJJ171011), and Innovation Special Fund for Individual Graduate Student of Jiangxi University of Finance and Economics (2020 Annual, No. 24).

References

- [1] Cisco, *Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2017–2022 White Paper*, 2019, <https://www.cisco.com/c/en/us/solutions/collateral/serviceprovider/visual-networking-index-vni/white-paper-c11-738429.html>.
- [2] Y. C. Hu, M. Patel, D. Sabella, N. Sprecher, and V. Young, "Mobile edge computing—a key technology towards 5G," *ETSI White Paper*, vol. 11, 2015.
- [3] K. Zhang, Y. Mao, S. Leng et al., "Energy-efficient offloading for mobile edge computing in 5G heterogeneous networks—efficient offloading for mobile edge computing in 5G heterogeneous networks," *IEEE Access*, vol. 4, pp. 5896–5907, 2016.
- [4] A. Ahmed and E. Ahmed, "A survey on mobile edge computing," in *Proceedings of the IEEE International Conference on Intelligent Systems and Control (ISCO)*, Coimbatore, India, 2016.
- [5] J. Liu, Y. Mao, J. Zhang, and K. B. Letaief, "Delay-optimal computation task scheduling for mobile-edge computing systems," in *Proceedings of the 2016 IEEE International Symposium on Information Theory (ISIT)*, pp. 1451–1455, Barcelona, Spain, July 2016.
- [6] Y. Mao, J. Zhang, and K. B. Letaief, "Dynamic computation offloading for mobile-edge computing with energy harvesting devices," *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 12, pp. 3590–3605, 2016.
- [7] T. X. Tran, A. Hajisami, P. Pandey, and D. Pompili, "Collaborative mobile edge computing in 5G networks: new paradigms, scenarios, and challenges," *IEEE Communications Magazine*, vol. 55, no. 4, pp. 54–61, 2017.
- [8] D. Wang, Y. Peng, X. Ma et al., "Adaptive wireless video streaming based on edge computing: opportunities and approaches," *IEEE Transactions on Services Computing*, no. 99, p. 1, 2018.
- [9] Z. Xiong, Y. Zhang, D. Niyato, P. Wang, and Z. Han, "When mobile blockchain meets edge computing," *IEEE Communications Magazine*, vol. 56, pp. 8–39, 2018.
- [10] Y. Liu, "2020 Index IEEE communications Surveys&Tutorials vol. 22," *IEEE Communications Surveys and Tutorials*, vol. 22, no. 4, pp. 1–16, 2020.
- [11] N. Barman, G. C. Deepak, and M. G. Martini, "Blockchain for video streaming: opportunities, challenges and open issues," *Computer*, vol. 53, p. 7, 2020.
- [12] D. C. Nguyen, P. C. Pathirana, N. Ding, and A. Seneviratne, "Blockchain for 5G and beyond networks: a state of the art survey," *Journal of Network and Computer Applications*, vol. 166, Article ID 102693, 2020.
- [13] X. Wang, C. Wang, X. Li, V. C. M. Leung, and T. Taleb, "Federated deep reinforcement learning for internet of things with decentralized cooperative edge caching," *IEEE Internet of Things Journal*, vol. 7, no. 10, pp. 9441–9455, 2020.
- [14] X. Jiang, F. R. Yu, T. Song, and V. C. M. Leung, "Resource allocation of video streaming over vehicular networks: a survey, some research issues and challenges," *IEEE Transactions on Intelligent Transportation Systems*, vol. 99, pp. 1–21, 2021.
- [15] Y. Wei, S. Zhou, S. Leng, S. Maharjan, and Y. Zhang, "Federated learning empowered end-edge-cloud cooperation for 5G HetNet security," *IEEE Network*, vol. 35, no. 2, pp. 88–94, 2021.
- [16] Y. Pan, N. Xiong, and J. Ren, "Data security and privacy protection for cloud storage: a survey," *IEEE Access*, vol. 8, pp. 131723–131740, 2020.
- [17] N. Zhang, C. Wu, Y. Wu, and N. N. Xiong, "An improved target tracking algorithm and its application in intelligent video surveillance system," *Multimedia Tools & Applications*, vol. 79, no. 23–24, pp. 15965–15983, 2020.
- [18] Y. Long, Y. Chen, W. Ren, H. Dou, and N. N. Xiong, "DePET: a decentralized privacy-preserving energy trading scheme for vehicular energy network via blockchain and K-anonymity," *IEEE Access*, vol. 8, pp. 192587–192596, 2020.
- [19] J. Wan, P. Zheng, H. Si, N. N. Xiong, W. Zhang, and A. V. Vasilakos, "An artificial intelligence driven multi-feature extraction scheme for big data detection," *IEEE Access*, vol. 7, pp. 80122–80132, 2019.
- [20] X. Xu, X. Zhang, H. Gao, Y. Xue, L. Qi, and W. Dou, "Be-Come: blockchain-enabled computation offloading for IoT in mobile edge computing," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 6, pp. 4187–4195, 2020.
- [21] J. Feng, F. Richard Yu, Q. Pei, X. Chu, J. Du, and L. Zhu, "Cooperative computation offloading and resource allocation for blockchain-enabled mobile-edge computing: a deep reinforcement learning approach," *IEEE Internet of Things Journal*, vol. 7, no. 7, pp. 6214–6228, 2020.
- [22] F. Guo, F. R. Yu, H. Zhang, H. Ji, M. Liu, and V. C. M. Leung, "Adaptive resource allocation in future wireless networks with blockchain and mobile edge computing," *IEEE Transactions on Wireless Communications*, vol. 19, no. 3, pp. 1689–1703, 2020.
- [23] Y. Dai, D. Xu, and K. Zhang, "Deep reinforcement learning and permissioned blockchain for content caching in vehicular edge computing and networks," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 4, pp. 4312–4324, 2020.
- [24] Q. Xu, Z. Su, and Q. Yang, "Blockchain-based trustworthy edge caching scheme for mobile cyber-physical system," *IEEE Internet of Things Journal*, vol. 7, no. 2, pp. 1098–1110, 2020.
- [25] M. Liu, F. R. Yu, Y. Teng, V. C. M. Leung, and M. Song, "Distributed resource allocation in blockchain-based video

- streaming systems with mobile edge computing,” *IEEE Transactions on Wireless Communications*, vol. 18, no. 1, pp. 695–708, 2019.
- [26] Y. Liu, F. R. Yu, X. Li, H. Ji, and V. C. M. Leung, “Decentralized resource allocation for video transcoding and delivery in blockchain-based system with mobile edge computing,” *IEEE Transactions on Vehicular Technology*, vol. 68, no. 11, pp. 11169–11185, 2019.
- [27] M. Liu, Y. Teng, F. Yu, V. Leung, and M. Song, “A mobile edge computing (MEC)-Enabled transcoding framework for blockchain-based video streaming,” *IEEE Wireless Communications*, vol. 99, pp. 1–7, 2020.
- [28] R. Zhang, F. R. Yu, J. Liu, T. Huang, and Y. Liu, “Deep reinforcement learning (DRL)-based device-to-device (D2D) caching with blockchain and mobile edge computing,” *IEEE Transactions on Wireless Communications*, no. 99, 2020.
- [29] R. Zhang, F. R. Yu, J. Liu, R. Xie, and T. Huang, “Blockchain-incentivized D2D and mobile edge caching: a deep reinforcement learning approach,” *IEEE Network*, vol. 34, no. 4, pp. 150–157, 2020.
- [30] X. Qiu, L. Liu, W. Chen, Z. Hong, and Z. Zheng, “Online deep reinforcement learning for computation offloading in blockchain-empowered mobile edge computing,” *IEEE Transactions on Vehicular Technology*, vol. 68, no. 8, pp. 8050–8062, 2019.
- [31] Z. Zhang, Z. Hong, W. Chen, Z. Zheng, and X. Chen, “Joint computation offloading and coin loaning for blockchain-empowered mobile-edge computing,” *IEEE Internet of Things Journal*, vol. 6, no. 6, pp. 9934–9950, 2019.
- [32] H. Cheng, Z. Xie, L. Wu, Z. Yu, and R. Li, “Data prediction model in wireless sensor networks based on bidirectional LSTM,” *EURASIP Journal on Wireless Communications and Networking*, pp. 1–12, 2019.
- [33] Y. Yang, N. Xiong, N. Y. Chong, and X. Défago, “A decentralized and adaptive flocking algorithm for autonomous mobile robots,” in *Proceedings of the 3rd International Conference on Grid and Pervasive Computing*, Kuming, China, May 2008.
- [34] Q. Zhang, C. Zhou, N. Xiong, Y. Qin, X. Li, and S. Huang, “Multimodel-based incident prediction and risk assessment in dynamic cybersecurity protection for industrial control systems,” *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 46, no. 10, pp. 1429–1444, 2015.
- [35] K. Huang, Q. Zhang, C. Zhou, N. Xiong, and Y. Qin, “An efficient intrusion detection approach for visual sensor networks based on traffic pattern learning,” *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 47, no. 10, pp. 2704–2713, 2017.
- [36] A. Shahzad, M. Lee, Y.-K. Lee et al., “Real time MODBUS transmissions and cryptography security designs and enhancements of protocol sensitive information,” *Symmetry*, vol. 7, no. 3, pp. 1176–1210, 2015.
- [37] W. Wu, Y. Gao, T. Zhou et al., “Deep reinforcement learning-based video quality selection and radio bearer control for mobile edge computing supported short video applications,” *IEEE Access*, vol. 7, pp. 181740–181749, 2019.
- [38] L. Lei, X. Xiong, L. Hou, and K. Zheng, “Collaborative edge caching through service function chaining: architecture and challenges,” *IEEE Wireless Communications*, vol. 25, no. 3, pp. 94–102, 2018.
- [39] C. H. Wei, Y. W. Hung, and F. L. Chin, “Q-learning based collaborative cache allocation in mobile edge computing,” *Future Generation Computer Systems*, vol. 102, pp. 603–610, 2020.
- [40] Z. Yang, Y. Liu, Y. Chen, and G. Tyson, “Deep reinforcement learning in cache-aided MEC networks,” in *Proceedings of the ICC 2019-2019 IEEE International Conference on Communications (ICC)*, IEEE, Shanghai, China, 2019.
- [41] C. Zhong, M. C. Gursoy, and S. Velipasalar, “Deep reinforcement learning based edge caching in wireless networks,” *IEEE Transactions on Cognitive Communications and Networking*, 2020.
- [42] M. C. Gursoy, C. Zhong, and S. Velipasalar, “Deep magnet reinforcement learning for cooperative edge caching,” *Machine Learning for Future Wireless Communications*, pp. 439–457, 2020.
- [43] L. Liu, H. Hu, Y. Luo, and Y. Wen, “When wireless video streaming meets AI: a deep learning approach,” *IEEE Wireless Communications*, vol. 27, 2019.
- [44] H. Cheng, Z. Su, N. Xiong, and Y. Xiao, “Energy-efficient node scheduling algorithms for wireless sensor networks using Markov Random Field model,” *Information Sciences*, vol. 329, no. 2, pp. 461–477, 2016.
- [45] H. Cheng, N. Xiong, A. V. Vasilakos, L. Tianruo Yang, G. Chen, and X. Zhuang, “Nodes organization for channel assignment with topology preservation in multi-radio wireless mesh networks,” *Ad Hoc Networks*, vol. 10, no. 5, pp. 760–773, 2012.
- [46] W. Wu, N. Xiong, and C. Wu, “Improved clustering algorithm based on energy consumption in wireless sensor networks,” *IET Networks*, vol. 6, no. 3, pp. 47–53, 2017.
- [47] J. Wang, Y. Ding, N. N. Xiong, W.-C. Yeh, and J. Wang, “GSCS: general secure consensus scheme for decentralized blockchain systems,” *IEEE Access*, vol. 8, pp. 125826–125848, 2020.
- [48] H. Cheng, Z. Xie, Y. Shi, and N. Xiong, “Multi-step data prediction in wireless sensor networks based on one-dimensional CNN and bidirectional LSTM,” *IEEE Access*, vol. 7, pp. 117883–117896, 2019.
- [49] H. Cheng, L. Wu, Y. Zhang, and N. Xiong, “Data recovery in wireless sensor networks using Markov random field model,” *ICACI*, pp. 706–711, 2018.
- [50] N. Xiong, X. Huang, H. Cheng, and Z. Wan, “Energy-efficient algorithm for broadcasting in ad hoc wireless sensor networks,” *Sensors*, vol. 13, no. 4, pp. 4922–4946, 2013.
- [51] C. Zhong, M. C. Gursoy, and S. Velipasalar, “A deep reinforcement learning-based framework for content caching,” in *Proceedings of the 2018 52nd Annual Conference on Information Sciences and Systems (CISS)*, March 2018.
- [52] Y. Zheng, D. Wu, Y. Ke, C. Yang, M. Chen, and G. Zhang, “Online cloud transcoding and distribution for crowdsourced live game video streaming,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 27, no. 8, pp. 1777–1789, 2017.
- [53] T.-Y. Huang, R. Johari, N. Mckeown, M. Trunnell, and M. Watson, “A buffer-based approach to rate adaptation,” *ACM SIGCOMM Computer Communication Review*, vol. 44, no. 4, pp. 187–198, 2015.
- [54] Y. Guo, F. R. Yu, J. An, K. Yang, G. Yu, and V. C. M. Leung, “Adaptive bitrate streaming in wireless networks with transcoding at network edge using deep reinforcement learning,” *IEEE Transactions on Vehicular Technology*, vol. 99, p. 1, 2020.
- [55] M. Chen, M. Ponc, S. Sengupta, J. Li, and P. A. Chou, “Utility maximization in peer-to-peer systems with applications to video conferencing,” *IEEE/ACM Transactions on Networking*, vol. 20, no. 6, pp. 1681–1694, 2012.
- [56] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, MIT Press, Cambridge, MA, USA, 2018.

- [57] V. Mnih, K. Kavukcuoglu, D. Silver et al., "Playing Atari with deep reinforcement learning," in *Proceedings of the NIPS Deep Learning Workshop*, Lake Tahoe, NV, USA, 2013.
- [58] V. Mnih, K. Kavukcuoglu, D. Silver et al., "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533, 2015.
- [59] W. Zheng and Y. Li, "Deep reinforcement learning-based collaborative video caching and transcoding in clustered and intelligent edge B5G networks," *Wireless Communications and Mobile Computing*, vol. 2020, Article ID 6684293, 16 pages, 2020.

Research Article

Towards Trustworthy IoT: A Blockchain-Edge Computing Hybrid System with Proof-of-Contribution Mechanism

Huan Dai ¹, Pengzhan Shi ¹, He Huang,² Ruyu Chen ¹ and Jun Zhao ¹

¹School of Electronic and Information Engineering, Suzhou University of Science and Technology, Suzhou 215000, China

²School of Computer Science and Engineering, Soochow University, Suzhou 215000, China

Correspondence should be addressed to Huan Dai; daihuanjob@163.com

Received 1 June 2021; Accepted 14 August 2021; Published 29 August 2021

Academic Editor: Jie Cui

Copyright © 2021 Huan Dai et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The emerging smart city is driving massive transformations of modern cities, facing the huge influx of sensor data from IoT devices. Edge computing distributes computing tasks to the near-edge end, which greatly enhances the service quality of IoT applications, that is, ultralow latency, large capacity, and high throughput. However, due to the constrained resource of IoT devices, currently, systems with a centralized model are vulnerable to attacks, such as DDoS from IoT botnet and central database failure, which can hardly provide high-confidence services. Recently, blockchain with a high security promise is considered to provide new approaches to enhancing the security of IoT systems. However, blockchain and IoT have obvious incompatibility, and low-capacity IoT devices can hardly be incorporated into blockchain with high computing requirements. In this paper, a blockchain-edge computing hybrid system (BEHS) is presented to make the adaptation of blockchain to edge computing and provide trustworthy IoT management services for a smart city. A novel extensible consensus protocol designed for proof-of-work, named proof-of-contribution (PoC), is proposed to regulate the data upload behaviors of nodes, especially the data upload frequency of IoT device nodes, so as to protect the system from attack about frequency. In order to secure the data privacy and authenticity, a data access control scheme is designed by integrating symmetric encryption with asymmetric encryption algorithm. We implemented a concrete BEHS on Ethereum, realized the function of PoC mechanism via smart contracts, and conducted a case study for smart city. The extensive evaluations and analyses show that the proposed PoC mechanism can effectively detect and automatically manage the behavior of nodes, and the time cost of data access control scheme is within an acceptable range.

1. Introduction

1.1. Motivations. The rapid advancement of the Internet-of-Things (IoT) technologies has greatly promoted the intelligent transformation of modern cities, making the realization of smart city getting closer [1, 2]. The extensive usage of IoT devices causes the storm growth of cloud traffic. Edge computing is an emerging computing paradigm, which decentralizes computing tasks to the near-edge end to improve the quality of smart city applications and services [3]. Meanwhile, the rise of IoT has also brought some security concerns to smart cities. Due to the constrained resources, IoT devices are vulnerable to attacks. In 2016, the Dyn data center was attacked by large-scale DDoS attack from IoT devices affected by botnet, resulting in a long time

interruption of relevant services and a large number of enterprise losses [4]. Moreover, systems with centralized model are vulnerable to central database failure and are not conducive to secure the data authenticity. In addition, lack of promised approaches of value transmission constrains the application scenario of IoT [5].

Cloud and edge computing also have some problems to be solved. For example, cloud platform that stores heterogeneous data is still not yet deviated from the essence of centralization, which leads to users overrelying on trust in cloud platforms for data access control. Furthermore, edge computing has the ability to continuously receive and process omnipresent data, which seems to be unremarkable in the field of data privacy protection. More importantly, the operation and maintenance of such a large yet centralized

IoT system requires an immense cost. These problems are restricting the development and progress of the Internet of things. Bitcoin [6], a decentralized cryptocurrency introduced by Nakamoto in 2008, provides a trustworthy method to transfer information in the untrustworthy environment, which is the first phenomenal application of blockchain. Owing to the distributed and digital trust properties of blockchain, the integration of blockchain into IoT architecture based on edge computing is a feasible and potential scheme [7–9].

Zhang et al. [10] designed a smart contract-based framework to investigate the access control issue in IoT systems. Huang et al. [11] developed a high-throughput industrial IoT blockchain system based on the principle of directed acyclic graph in distributed ledger technology (DLT). Pan et al. [12] proposed an EdgeChain framework to link the resource of edge servers with IoT objects by a coin system based on smart contracts. However, there are challenges remaining unsolved when integrating blockchain with edge computing. Numerous and heterogeneous IoT devices make smart contract undertake higher complexity and storage cost, which leads to inefficiency of smart contract-based IoT management and can hardly meet the requirement of high throughput of IoT systems. Due to the constrained resource, IoT devices can hardly adapt to consensus algorithms with high computational complexity and large storage requirement, such as proof-of-work [13] and proof-of-stake [14], which makes IoT devices fall out system supervision and become vulnerabilities. Additionally, the transparency of blockchain makes it difficult to protect data privacy, which is contrary to the requirements of IoT systems. Even in the permissioned chain, data is not always intended to be disclosed to all permission participants. So far, there is a lack of blockchain-based scheme designed specifically for the IoT devices.

1.2. Related Work. The conventional IoT systems have achieved improvements in computing and storage capabilities based on cloud computing. However, IoT devices, as a large number of writers in the system, cannot verify the integrity of stored data. The questionable credibility of IoT devices and the complexity of the network also pose challenges to information security and data privacy of large-scale smart city systems. While centralized systems have strong performance, it is difficult to be applied in IoT scenarios due to the vulnerability of single point of failure and zero-tolerance of malicious writers. Instead, blockchain, a special distributed ledger technology that sacrifices performance in exchange for trust and availability, is considered a new solution. However, the performance of blockchain is difficult to meet the requirements of IoT with massive data. Because of this, a lot of works have been done to improve the scalability of consensus algorithm or blockchain on the premise of ensuring the security. Biswas et al. [15] present a novel lightweight proof of block and trade (POBT) consensus algorithm for IoT blockchain and its integration framework, allowing the validation of trades as well as blocks with reduced computation time. They proposed a new

allocation mechanism to reduce the memory requirements of IoT nodes. Viriyasitavat et al. [16] analyzed the pressure and risks of the quality of service (QoS) in the IoT and integrated the blockchain technologies (BCT) with a multi-agent approach to ensure the reliability of real-time data and achieve the measurement of QoS in the IoT environment. Guo et al. [17] constructed collaborative mining network (CMN) to execute mining tasks for mobile blockchain, which solves the problem that IoT mobile devices cannot afford the high computing cost of blockchain due to the limitations of communication and computing.

Wang et al. [18], especially, designed a trust consensus scheme for IIoT; it can be implemented on the state-of-the-art PoX consensus protocols. The reputation module of this scheme is equipped with an incentive mechanism; the participants will be motivated to make honest behavior and contribution for network. However, IIoT devices are defined as nodes in the blockchain network, which overestimates the storage and computing capacity of IIoT devices. Song et al. [19] proposed a proof-of-contribution consensus mechanism for intellectual property protection, which quantifies various behaviors and actions of nodes into specific contribution values. When the current state of system meets the conditions for generating a new block, nodes are sorted by their contribution values, and the node with the highest values will become the new bookkeeping node. However, although assigning contribution values to each node can effectively improve bookkeeping credibility, overreliance on contribution will aggravate the centralization.

1.3. Contributions. To address the aforementioned challenges, we proposed a novel blockchain-edge computing hybrid system (BEHS) to provide trustworthy IoT service for smart cities. The core idea of BEHS is to integrate edge computing with permissioned chain to enhance the security of IoT system while ensuring efficiency. Considering the system scalability, it is designed to have multilayers and multiple modules, which can run on different IoT systems, for example, smart home, smart industry, and smart transport. A novel proof-of-contribution consensus mechanism is proposed to regulate the behavior of nodes, especially IoT device nodes, securing the system from malicious attacks. A data access control scheme, which integrates the symmetric cryptography with the asymmetric cryptography, is also presented to secure the data privacy and authenticity during the communication. As a short summary, our main contributions of this paper include the following:

- (1) A novel framework integrates permissioned chain with edge computing, providing a decentralized model for IoT.
- (2) A proof-of-contribution (PoC) consensus mechanism is developed to provide a trustworthy management method for the nodes in IoT systems.
- (3) A data access control scheme is designed to realize the directional transmission and the privacy protection of IoT data.

- (4) We implemented a concrete system on Ethereum and conducted experiments to evaluate the system.

The remaining of this article is organized as follows: Section 2 presents the overview design of BEHS. Section 3 introduces a concrete BEHS on Ethereum platform. The evaluation is discussed in Section 4. The conclusions are discussed in Section 5.

2. Blockchain-Edge Computing Hybrid Systems for Trustworthy IoT

In this section, we introduced the proposed system with its key modules. The overview framework design of the system is presented first, and then, we introduced the PoC consensus mechanism and the data access control scheme in detail.

2.1. Overall Framework Design. The system framework is built on permissioned-chain and edge computing. As depicted in Figure 1, it can be divided into two layers with four essential modules. Infrastructure layer includes sensing device, blockchain, and edge server modules, which support the system environment and functionality. Application layer implements specific services for users, which commonly relies on the cloud to realize its function.

Multiple types of sensing devices, edge server groups in multiple regions, and blockchain key approaches form the infrastructure layer. Each node has a unique identity, a specific address, and a pair of public/private keys. A private peer-to-peer (P2P) network [20] is set up for communication, where all nodes can discover each other, transmitting and broadcasting transaction information.

Sensing devices collect and collate the samples data measured from physical environment. The data will be uploaded to several nearby edge servers at the same time, and edge servers will broadcast the data throughout the whole network, thus adding the data to the transaction pool, waiting for edge servers to pack. In this way, the system separates the one-to-one subordination relationship between sensing device and edge server. Every sensing device can be a stand-alone node, peer-to-peer with the edge server and constrained by the system rules.

Edge servers have powerful computing and network resources, providing the calculation power for generating new blocks and securing system consistency. The data from sensing devices will be stored into blocks, and the generation of a new block should contain the hash value of its previous block. Thus, blocks are stored in a chain structure to form the ledger. Once formed, it is hard to change any part of the ledger. Every edge server stores a real-time updated backup of the ledger locally to make the distributed storage of data in the system.

Blockchain supports vital security functions, including consensus mechanism and encryption algorithm. Consensus mechanism ensures the consistency of the ledger stored in edge servers. We considered that the regulatory function of sensing devices should be included in the consensus mechanism, which is the key to the separation of edge

devices from the subordinate relationship between servers. From this, we design a novel consensus mechanism, named proof-of-contribution (PoC).

Encryption algorithm and digital signature algorithm secure the communication between nodes. Data processed by encryption algorithm is usually difficult to be cracked in blockchain system, but while ensuring the security, it also undermines the convenience of data sharing. The digital signature algorithm can effectively ensure the integrity, credibility, and nonrepudiation of data, which is one of the reliable technical means of data sharing. Therefore, in the application scenario of smart city, we designed the data access control scheme that integrates encryption algorithm and digital signature algorithm to balance the requirements of security and data sharing.

Cloud is the interface of services for users, such as visualized analysis, device management, and privacy protection. Its implementation commonly relies on website platform, applet of WeChat, apps, and so on. Moreover, the system retains the valuable token mechanism, which is designed as a value container. It has a novel function: digitally incentivize the contribution and loyalty behavior of each node for the system. This mechanism encourages devices to upload timely and authentic data and stimulates the participation of merchants and other stakeholders to promote the development of IoT. In addition, the token system gives the ability to transfer value between entities and expands the application of IoT in economy related scenarios.

2.2. Proof-of-Contribution Mechanism. In this subsection, we present a novel consensus mechanism, named proof-of-contribution (PoC), which can synchronize the ledger and regulate the behavior of nodes, that is, sensing device node (SDN) and edge server node (ESN). PoC is inspired by PoW mechanism, which has been upgraded to adapt to IoT systems. We considered behaviors that benefit the system's services as contributing to the system. Sensing device contributes to the system by uploading data, and edge server contributes to the system by mining block. Any contribution will be recorded in the block, forming the system's ledger, and PoC rewards the contributors for encouraging their behaviors.

2.2.1. Edge Server. The edge server node (ESN) is the miner. Any behavior of nodes will be spread throughout the system and be added to the local transaction pool in every ESN. ESNs pack the behaviors in the transaction pool as an incomplete block and solve a hash puzzle to generate the block and log these behaviors. By inputting the information in a block into secure hash algorithm (SHA) function, such as SHA256 [21], the hash value of the block can be obtained. When the previous block hash value, Merkle root of the behaviors, timestamp, and nonce are input to SHA function, and the output is within the target range, the hash puzzle is solved, and the block is packed. If other ESNs verify that the block is correct and first published, the block will be accepted as the latest block in the ledger of the system.

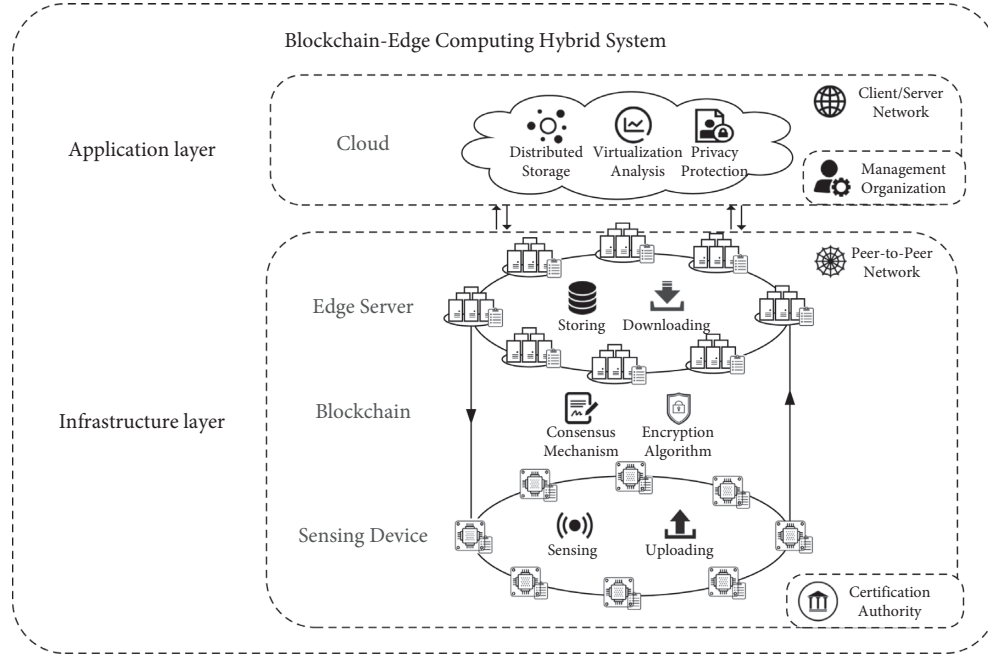


FIGURE 1: Framework design of the blockchain-edge computing hybrid system.

ESNs store the verified blocks locally. Since each block contains its previous block hash, the blocks are stored with chain structure in each ESN, thus making the distributed ledger. Due to the chain storage structure and large computing power for solving the hash puzzle, once the ledger is formed, it will be hard to tamper with the ledger's content. When a new block is verified and added into the distributed ledger as the latest block, PoC will reward the ESN that contributes to the block generation. Therefore, the structure of the block is designed to contain the address of contributors, as shown in Table 1.

From the aforementioned block verification scheme, the ESN with a higher computational power of hash might have a higher chance to find the correct nonce and obtain the reward. The probability P_e that the ESN e gets the reward of generating a new block with N ESNs is

$$P_e = \frac{H_e}{\sum_{k=1}^N H_k} \cdot D_e \cdot \sigma, \quad (1)$$

where H_e is the hash computational power of e , H_k is the hash computational power of k , D_e is impact factor of the density of ESNs adjacent to e in the network topology, and σ is the factor weight of the impact of time consumption. Due to the time consumption of message propagation in the system, ESNs close to the SDNs will receive behavior messages first, which means that it will start searching for nonce earlier than the ESNs far away from the SDNs. Therefore, the number of ESNs in the region, where ESNs are located, has a great impact on the opportunity to obtain rewards of mining, which conforms to the concept of edge computing. This makes ESNs with low computing power still have a relatively reasonable chance to get rewards through mining, effectively preventing ESNs with strong computing power from combining to monopolize rewards and opportunities of block generation.

TABLE 1: Block structure designed for BEHS.

Block header	
Number	Block height number
Timestamp	Creation time of the block
Hash	Hash of the block
ParentHash	Hash of the previous block
Nonce	Nonce
ESN	Address of ESNs
SDN	Address of SDNs
Difficulty	Mining difficulty value
Size	Length of the block in bytes
Block body	
Transaction	List of included behaviors

The computing resources of all the ESNs in a region can be considered a computing pool. When the computing resources overflow the pool, as the number of ESNs increases, the chance of each ESN to get a reward through mining will be reduced. Therefore, the mutual competition between ESNs in the same region will lead to the renewal and elimination of ESNs, which will allow the number of ESNs to be adapted to the number of SDNs. This also promotes the competition of interest organizations, which are the most innovative role in smart cities and provide fresh blood for the development of IoT industries.

2.2.2. Sensing Server. We defined that each sensing device node (SDN) s has a property of credit value C_s and a property of subcategory value r . PoC will dynamically evaluate the credit score of the SDN based on its past behaviors. The normal behaviors, that is, the SDN obeying the prefixed rules to upload data, will increase its credit score, while the abnormal behaviors, for example, the SDN uploading data at an incorrect

time interval or format will reduce its credit score. When a block is accepted by the system, the SDN contributing to the block will be rewarded. SDNs with high credit score are considered honest and have relatively high rewards. On the contrary, SDNs with low credit score will receive relatively less rewards. When the credit score of a SDN falls below the system threshold, it will be considered as a malicious attacker and be removed from the system.

Before giving the detailed mechanism of PoC for SDNs, we first introduced the possible existing abnormal behaviors of SDNs in the system.

- (1) Extra gain: in order to gain more reward, a “greedy” SDN wants to compete inappropriately for rewards, for example, arbitrarily shortening the upload interval. This will lead to unfair reward competition among SDNs and result in system resource redundancy.
- (2) Lazy strike: when a SDN fails to finish the work in time, it will be considered as a “lazy” device and stop contributing to the system. Although it does not threaten the security of the system, it affects the normal operation of related services in the system, which can reduce the service quality of the system.
- (3) Malicious attack: a “malicious” SDN would want to monopolize the upload authority or prevent others from uploading. It sacrifices itself to disrupt the normal operation of the system, for example, uploading at an ultrahigh frequency. This will cause network congestion and waste system resources, making the system unable to handle normal transactions.

Thus, according to the past behavior of s , the C_s can be denoted as

$$C_s = \delta_1 \cdot C_s^N + \delta_2 \cdot C_s^A, \quad (2)$$

where C_s^N represents the score of normal behaviors, C_s^A represents the score of abnormal behaviors, and δ_1 and δ_2 represent the system sensitivity to normal behaviors and abnormal behaviors, respectively, which can be adjusted dynamically to distribute the weight of these two parts according to application requirements.

C_s^N evaluates the quality of work completed by s , which is positively related to the upload interval Δt . When the interval is between $(2 - \alpha_s)\bar{t}_s$ and $\alpha_s\bar{t}_s$ (α_s is the preset parameter of the reasonable range of the specified upload interval \bar{t}_s , which is determined by the subcategory r of s), the behavior will be regarded as normal. Thus, C_s^N can be defined as

$$C_s^N = \sum_{i=1}^{K_s^N} \frac{1}{\eta^{|\bar{t}_s - \Delta t|} \cdot (t - t_i)^{\zeta_1}} (2 - \alpha_s), \quad \bar{t}_s > \Delta T > \alpha_s \bar{t}_s, \quad (3)$$

where K_s^N represents the total number of normal behaviors conducted by s , t represents current time, t_i represents the time point of the i th behavior conducted by s , Δt represents the time interval of i th behavior and $(i - 1)$ th behavior conducted by s , η represents the sensitivity factor of the difference between rated interval and actual interval, and ζ_1

is the impact index of time on the credit score of normal behaviors.

As described in (3), we can observe that the credit score of normal behaviors of a SDN is related to the quality of its work completion, that is, the upload interval. Besides, as time goes on, the influence of past behaviors on the credit score will decrease.

C_s^A is negatively related to the upload interval, which can be defined as

$$C_s^A = - \sum_{i=1}^{K_s^A} \frac{|\bar{t}_s - \Delta t|}{(t - t_i)^{\zeta_2} + \kappa}, \quad \Delta t \leq \alpha_s \bar{t}_s \cup \Delta t \geq (2 - \alpha_s) \bar{t}_s, \quad (4)$$

where K_s^A represents the total number of abnormal behaviors conducted by s , ζ_2 is the impact index of time on the credit score of abnormal behaviors, and κ is the constraint parameter, which can adjust the range of the impact of abnormal behaviors.

As described in (4), we can observe that when a behavior is judged as an abnormal one, its situation will also be determined by its interval time. From (2), we can observe that the normal behaviors will increase the credit score, and the abnormal behaviors will reduce the credit score. Moreover, with the passage of time, the influence of the past behavior on the score will gradually decrease.

After the behavior is recorded in the ledger, PoC rewards the corresponding SDN based on C_s by

$$P_s = \lambda_r \cdot \bar{P} \cdot \left(\frac{\xi^{C_s} - \xi^{-C_s}}{\xi^{C_s} + \xi^{-C_s}} + 1 \right), \quad (5)$$

where \bar{P} is the preset reward of the ESN for every time finishing their work, ξ is the sensitivity of the reward P_s to the credit score C_s , and λ_r represents the weight between the reward of ESNs and the reward of the r type SDNs, which can be adjusted according to the management requirements of different types of SDNs. The concrete setting of these parameters will be discussed in Section 4.1.

As described in (2) and (5), when abnormal behaviors happened, P_s will decrease timely according to the decrease of C_s , while when a normal behavior happened, P_s will increase timely according to the increase of C_s . Moreover, as the value of C_s increases, P_s will never be higher than the upper limit, that is, $2\lambda_r\bar{P}$. A SDN with continuous abnormal behaviors in a time unit will cause the sharp decline of C_s . When C_s is lower than the preset threshold of the system, the SDN will be considered as a malicious attacker, and it will be temporarily removed from the system.

2.3. Data Access Control Scheme. Since every behavior message needs to be broadcasted and transmitted many times throughout the system, it is essential to ensure the data authenticity during the communication. Moreover, behavior messages come mostly from IoT devices in smart cities, which involve the privacy of IoT users. In order to protect the data authenticity and privacy in the system, we designed the data access control scheme (DACs).

From the aforementioned framework design, we know that BEHS is built on permissioned blockchain, and every node holds a pair of asymmetric keys (Pk, Sk). The message encrypted by Sk can only be decrypted by the corresponding Pk . The digital signature algorithm [22] uses this method to realize the directional transmission and nontampering of messages, but the messages can be exposed during transmission, and the privacy can hardly be protected.

Symmetric encryption is a lightweight scheme with high efficiency, and asymmetric encryption can provide a secure distribution way for symmetric keys [11]. Thus, we considered integrating the symmetric and asymmetric keys to encrypt privacy data and control data access in IoT devices. The encryption process of DACS can be denoted as

$$\text{encryption}(M) = \begin{cases} \text{ENC}_{Sk_s}\{\text{SHA}\{\text{ENC}_K(M)\}\}, \\ \text{ENC}_K(M), \end{cases} \quad (6)$$

where Sk_s is the secret key of the sender, K is the symmetric key generated by sender, ENC is the abbreviation of encryption, ENC_{Sk_s} represents the encryption by Sk_s , ENC_K represents the encryption by K , M denotes the message to be transmitted, and $\text{SHA}()$ represents generating the summary information leveraging secure hash algorithm. The receiver can decrypt the $\text{ENC}_{Sk_s}\{\text{SHA}\{\text{ENC}_K(M)\}\}$ by the public key Pk_s of the sender and verify the authenticity of the message by comparing $\text{SHA}\{\text{ENC}_K(M)\}$ with $\text{ENC}_K(M)$. The sender stores the key locally and send it to its owner. If a user wants to get the data, it needs to request the corresponding key from the owner of the data to decrypt the data. The distribution of K can be denoted as

$$\text{distribution}(K) = \text{ENC}_{Sk_s}\{\text{ENC}_{Pk_U}\{K\}\}, \quad (7)$$

where Pk_U is the public key of the user, and ENC_{Pk_U} represents the encryption by Pk_U . DACS utilizes user's public key to ensure that only the target user can decrypt the correct symmetric key.

According to the aforementioned scheme, the message format is designed for the system, as listed in Table 2. Every message is directed and contains the address of sender and receiver. The type field represents the type of content contained in the message, including data uploading, currency trading, and smart contract calling. If the message contains an upload behavior, the value and contract code fields can be blank, and the uploaded data is stored in payload field. For calling the smart contract, the payload field can be blank, and for the simple currency transaction, the payload and contract code fields are both blank.

Worthy of note is that three different types of timestamps exist in our system, respectively, recorded by sensing device, edge server, and blockchain. The combination of three timestamps creates a complete timeline for each behavior message uploaded by device. This timeline has the ability to trace and review behavior messages and preclude devices from uploading duplicate data to defraud credit value and rewards. The following is a detailed explanation of three timestamps:

- (1) Creation timestamp of the behavior: t_s . When a sensing device collects a sufficient amount of data, it

TABLE 2: Format of behavior message.

Header	
Type	Type of the behavior
From	Sender's address
Order	Number of behaviors from the sender
To	Receiver's address
Value	Number of currencies
Timestamp	Creation time of the behavior
<i>Payload</i>	
Data ₁ , data ₂ , ..., data _n	
<i>Contract code</i>	
Variable ₁ , variable ₂ ,..., variable _n	

will send data to the edge server in the format of behavior message.

- (2) Reception timestamp of behavior message: t_e . If the edge server monitors a behavior message sent by a sensing device, it will record the timestamp of that moment. In the event that the edge server detects t_s is too close to t_e or even later than t_e , it will refuse to store and package this behavior message.
- (3) Creation timestamp of the block: t_b . While cloud or edge server packs the accumulated transactions to generate a new block, the system will save the timestamp of the block generation time based on specification requirements of the block structure.

The flowchart of DACS is shown in Figure 2, which can be divided into two parts. Part 1 includes encryption, uploading, and storage of data. Part 2 includes request and distribution of symmetry key and decryption of data. The nonce is a random check code. If the receiver returns the right nonce, we will consider the receiver has decrypted the message correctly. When a sensing device submits data to an edge server, it utilizes its asymmetric keys to sign the data in payload field. The consistency of the summary and content of the data ensures the authenticity of the message. In this way, all messages in the system are traceable, and edge servers cannot tamper with behavior messages of sensing devices during broadcasting. For private data that need to be protected, the sensing device will generate a random symmetric key to encrypt the data, store the key locally and then distribute it to the administrator. Thus, the data stored in the ledger are encrypted by symmetric keys, and getting the access to the data requires obtaining the corresponding symmetric key. If a user wants to get the access of the data, he needs to request the corresponding key from the administrator to decrypt the data. If agreed, the sensing device will distribute the corresponding symmetric key by utilizing user's public key. Thus, the administrator or the node itself can protect the privacy of the data through controlling the access of the data.

3. System Implementation

In this section, we present the detailed implementation of the proposed BEHS, following by the evaluation of its performance.

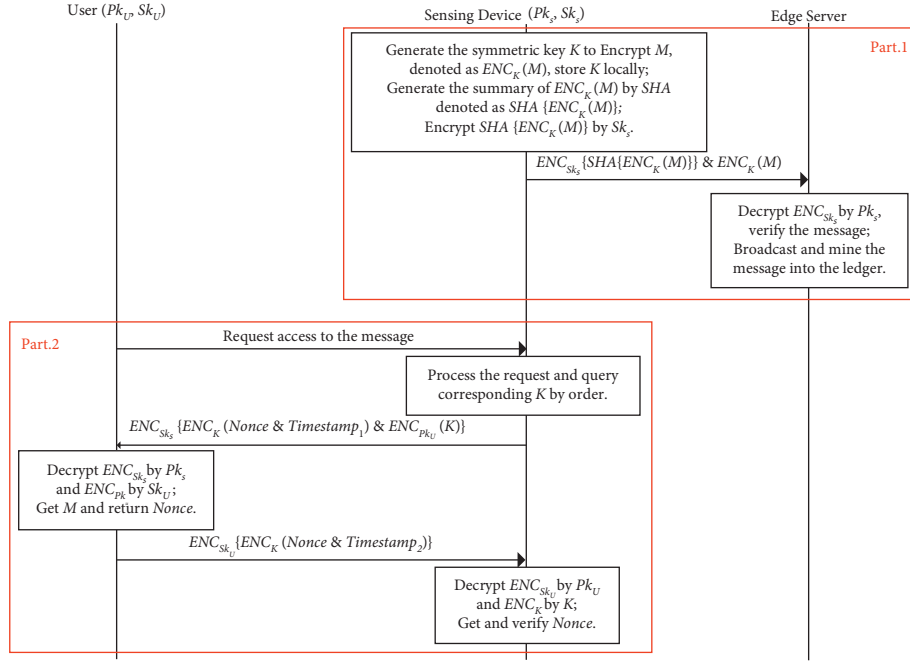


FIGURE 2: Flowchart of data access control scheme.

3.1. Ethereum-Based BEHS for Smart City

3.1.1. Edge Server. We installed Go-Ethereum on each of three clouds and built a consortium chain network by initializing the same configuration of the genesis.json file [23]. The configuration of the cloud is listed in Table 3, and some modules have already been embedded in Ethereum for providing interface, as summarized in Table 4. Cloud has the same status as edge server in distributed data processing, which packs transactions from the transaction pool and provides the computation support for mining new blocks. The block includes the block header and body, which will be synchronized locally to each cloud. Block generation requires a certain time interval to ensure the consistency of nodes, and the capacity of transaction payload for containing data is at most 1024 bytes. Therefore, compared with the ideal state of BEHS, the performance of Ethereum-based BEHS is significantly limited.

3.1.2. Sensing Device. We chose several common monitoring devices as sensing device nodes, including smoke, lampblack, and current devices. Each sensing device consists of an ARM Cortex-M3-based 32-bit processor named STM32F103VCT6 and a SIM7020 C NB-IoT module operated by China Telecom. We deploy multiple laptops as the agent to help sensing devices connect to Ethereum network. Each laptop utilizes the Geth client of Ethereum and connects the network as a light node, which only needs to store the block header and verifies blocks via Merkle Proof [6]. Each sensing device communicates with its agent through UDP protocol of NB-IoT gateway and uploads data through the RPC interface of the agent. In this case, the rewards of sensing devices cannot be directly distributed to their accounts, being held by the account of their agent. Additionally, we implemented the data access control scheme by

TABLE 3: Configuration of cloud.

Attributes	Configuration
Processor	Inter(R) core(TM) i5-3470 3.20 GHz
Memory	4 GB
OS	Windows 7
Database	Oracle
Disk	1 TB

TABLE 4: Modules in Ethereum-based BEHS.

Module	Technology
RPC interface	RPC APIs of Go language
Smart contract interface	Ethereum virtual machine
Application interface	Web3 protocol of JavaScript

integrating ECDSA based on the secp256k1 elliptic curve with AES symmetric encryption algorithm in each sensing device. Each piece of data uploaded from sensing devices is encrypted by a randomly generated symmetric key of 16 bytes. The key is stored locally on the device and periodically synchronized to its agent.

3.2. Implementation of PoC via the Smart Contract. We implemented PoC on Ethereum-based BEHS via multiple smart contracts. An entry contract is the entrance for receiving data from sensing device, and a judgment contract is responsible for evaluating credit scores and rewarding contributors.

3.2.1. Entry Contract. A lookup table containing the identity information of sensing devices is established in entry contract, as shown in Table 5, in which each row contains the following information of each sensing device:

TABLE 5: Illustration of the lookup table.

DecAddress	AgeAddress	DecType	Order	CreScore	AccAuthority
0xs35bpjk3...	0x6b6cad3c...	Smoke	35	2.1	True
0xdj92j29e...	0x2jk43lv4...	Lampblak	108	-20	False
0xd31f60gl...	0x2816b60s...	Current	0	0	True

- (1) DecAddress: address of the sensing device
- (2) AgeAddress: address of the agent
- (3) DecType: type of the sensing device
- (4) Order: number of messages from the sensing device
- (5) CreScore: credit score of the sensing device
- (6) AccAuthority: access authority of the sensing device

Thus, the entry contract can store the record of every behavior and the access authority of each sensing device. In addition, the entry contract provides the following application binary interfaces (ABIs) to maintain the lookup table:

`deviceRegister()`: this ABI receives the identity information of a new sensing device and registers its information into the lookup table.

`deviceUpdate()`: this ABI receives the new identity information of an existing sensing device and updates its information in the lookup table.

`deviceDelete()`: this ABI deletes the existing identity information of a sensing device from the lookup table.

`accControl()`: this ABI receives the credit score from judgment contract, controls the access authority, and updates the related information of the lookup table, especially the fields of CreScore and AccAuthority.

We noticed that only nodes with the authorized address can register, update, and delete the sensing device. The entry contract also has `subData()` ABI for receiving data from sensing devices; `subData()` ABI will call ABIs of judgment contract for judging the credit score and transact with the judgment contract to log the data.

3.2.2. Judgment Contract. The judgment contract implements a behavior evaluation method. When the data from a sensing device is logged in the block, judgment contract will reward the sensing device according to its credit score by transacting with its agent. A timestamp list is established for recording every behavior of each sensing device. The judgment contract provides the `timUpdate()` ABI for updating the list and the `timQuery()` ABI for querying timestamps from the list.

Based on the proposed PoC mechanism, we implemented the `creEvaluation()` ABI in judgment contract, as in Algorithm 1. It evaluates the credit score according to the inputs of the source address, type, timestamp and hash of a behavior, and returns the result. The evaluation of credit score is from lines 1 to 7 by (2), and the distribution of reward is from lines 8 to 17 by (5). The event in line 18 is used to return the results of updating the credit score and the reward distribution.

The detailed workflow of PoC mechanism is shown in Figure 3, which can be described in the following steps:

- (1) Edge servers initialize the private chain based on Go-Ethereum and create account. Sensing devices and their agents create accounts by the integrated encryption scheme and connect to the blockchain network.
- (2) Agents deploy entry and judgment contract on the chain, and sensing devices call entry contract to register their identity information.
- (3) Then, sensing devices upload data through the RPC port of its agent and call ABIs in entry contract for recording this behavior.
- (4) After that, when the behavior is packed by an edge server into a block and accepted by the system, the edge server will get the reward for mining, and judgment contract will evaluate the credit score of sensing devices and reward them.

4. Evaluation

In this section, we start by introducing the specific parameters setting of Ethereum-based BEHS, and then, we evaluate the performance of the system, including the effectiveness of PoC and the cost of DACS.

4.1. Parameters Setting. According to (2), the weight of the impact of normal and abnormal behaviors on credit score is 1 : 1, so we set $\delta_1 = 1$ and $\delta_2 = 1$. According to (3), due to the short time between the production and judgment time of each behavior message, we set $\zeta_1 = 1/2$ to constrain the impact of time on credit score. The upload period of sensing devices is set to 30 s, and \bar{t}_s is thus set to 30. When η is set to 1, the credit score of normal behavior will be fixed. In order to motivate sensing devices to provide high-quality services, we set η to 2. The timestamps of behaviors will be stored in the list of judgment contract; thus, \bar{t}_s and K_s^N can be calculated. A large tolerance of abnormal behaviors is beneficial to the effectiveness evaluation of the system, so we set α_s to $2/3$.

For a sensing device with abnormal behaviors, PoC will limit its future rewards in several cycles. The negative impact of the abnormal behavior will gradually decrease over time but cannot be erased. Therefore, the reward of node with abnormal behavior will always be less than node without abnormal behavior with the same performance. According to (4), we set the decrease rate of the influence of abnormal behavior to be the same as that of normal behavior, $\zeta_2 = \zeta_1 = 1/2$. κ is set to 2, which adjusts the impact range of abnormal behaviors according to that of normal behaviors.


```

Input: address, type, timestamp, hash
Output: result (update, reward)
Require: result.update ← False, result.reward ← False.
Create a timestamp array timestampArray[].
timestampArray ← timQuery (address)
get Cs (address, timestampArray) using (2)
create an Entry Contract instance entry
if entry.accControl(address, Cs) is captured then
    result.update ← True
end if
while true do
    check the transaction receipt
    if the block containing the behavior is generated then
        get Ps(Cs) using (5)
        create a transaction(address, Ps)
        launch the transaction.
        result.reward ← True
        break.
    end if
end while
return result (update, reward)

```

ALGORITHM 1: creEvaluation() ABI.

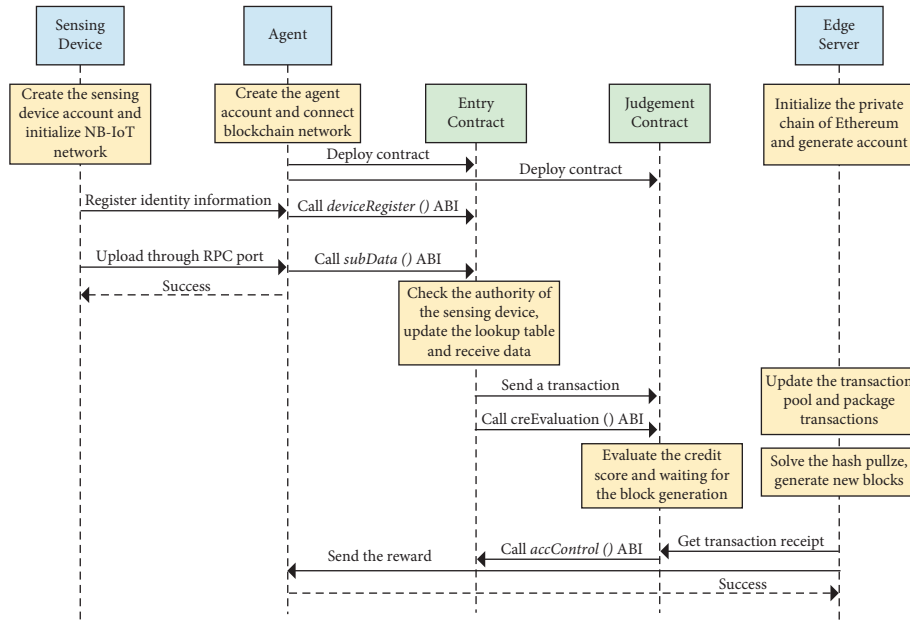


FIGURE 3: Flowchart of PoC via the smart contract.

If a larger range is desired, we can set it smaller. K_r^A can also be calculated from the list in judgment contract.

Based on the previously mentioned setting, PoC can evaluate and regulate behaviors of sensing device according to its credit score. According to (5), there are three parameters λ_r , ξ , and \bar{P} . Every time a new block is mined, PoC will reward 5 Ether to the corresponding miner, so \bar{P} is thus set to 5. The weight between the reward of edge server and sensing device can be adjusted according to the needs, where we set $\lambda_r = 1$ in the experiment. ξ is set to 2, which is not a large sensitivity level. However, this value can be adjusted if needed.

4.2. Effectiveness Proof-of-Contribution. To present the effectiveness of PoC, we set sensing devices in different states for simulating different types of behaviors. Figure 4 shows the results of credit score changes based on behaviors of sensing device. The x -axis represents the timeline, containing multiple Δt . The period of normal behavior is set to 30 s, and the abnormal behavior is divided into the malicious behavior and lazy behavior. The period of the malicious behavior is set to 10s, and the lazy behavior will stop working for 130 s. The y -axis represents the value of credit score, with three curves representing normal behavior score,

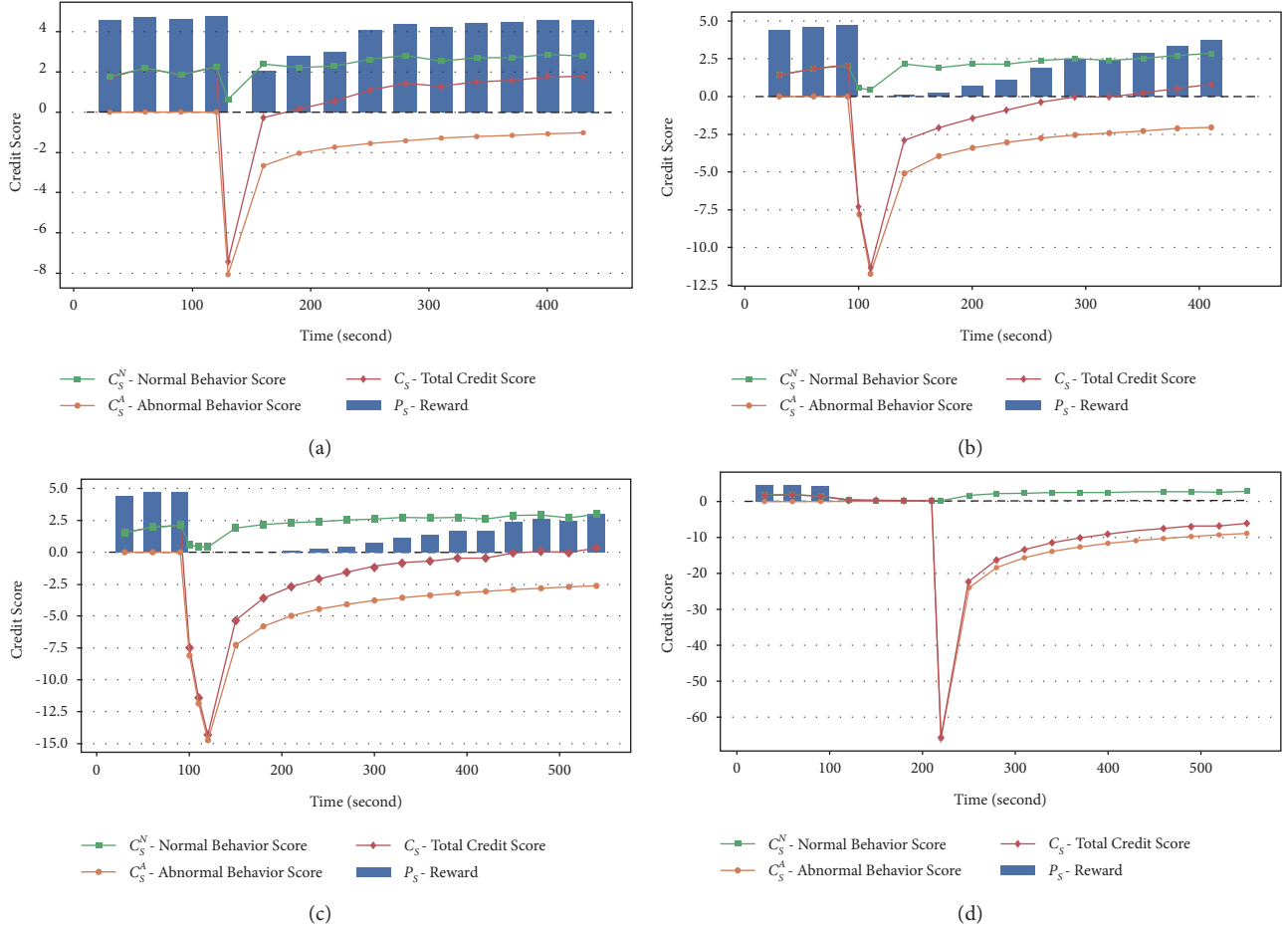


FIGURE 4: The credit score changes based on behaviors of sensing devices. (a) When a malicious behavior happens. (b) When malicious behaviors happen twice. (c) When malicious behaviors happen three times. (d) When a lazy behavior happens.

abnormal behavior score, and total credit score, respectively. The reward is also denoted according to the changes of the total credit score.

As can be seen from Figure 4(a), when time is at 110s, the sensing device conducts a malicious behavior. C_s^N has a sharp decline according to (4), C_s also has a sharp decline according to (2), and P_s is decreased to 0. After the malicious behavior, the sensing device continues to behave normally. After several normal behavior periods, its total credit score gradually recovers but is lower than the average before the malicious behavior. In Figure 4(b), when the sensing device commits two consecutive malicious behaviors, it will be punished more severely. After these two malicious behaviors, the next normal behavior will lose its reward, and more normal behaviors are required to recover the total credit score. The average reward after recovery is lower than the average reward after one abnormal behavior. Certainly, the system will not endlessly tolerate a node with too much negative behavior. When a node's C_s falls below the threshold at a certain moment, it will be kicked out of the network. Figure 4(c) shows the change of credit score and reward of the sensing device conducting three times consecutive malicious behaviors. In addition to the malicious behaviors not being rewarded,

the next three normal behaviors will also not be rewarded. The recovery period of credit score is longer and the average reward is lower than the first two cases. As can be seen from Figure 4(d), the sensing device stops uploading data after 90s and then starts uploading data normally at 220s. The credit score of the sensing device continuously decreases during the shutdown. Since the system cannot capture lazy behaviors from the sensing device, no abnormal behavior score accumulates. When the sensing device resumes uploading data, the first behavior is considered as a lazy behavior, and the credit score plummets. Due to the long shutdown period, the sensing device needs to reduce the impact of this abnormal behavior through more time of normal behavior.

To further study the relationship between reward and behavior cycle, we analyzed the changes of the reward based on different uploading periods, that is, 20 s, 22 s, ..., 34 s, 36 s, and the result is shown in Figure 5. We can observe that, with the increase of the gap between the actual period and the preset period, the credit score gradually decreases. The change of reward is more sensitive to the uploading period than the change of credit score. Sensing devices cannot get more rewards by reducing their uploading period, which encourages sensing devices to complete their work more honestly.

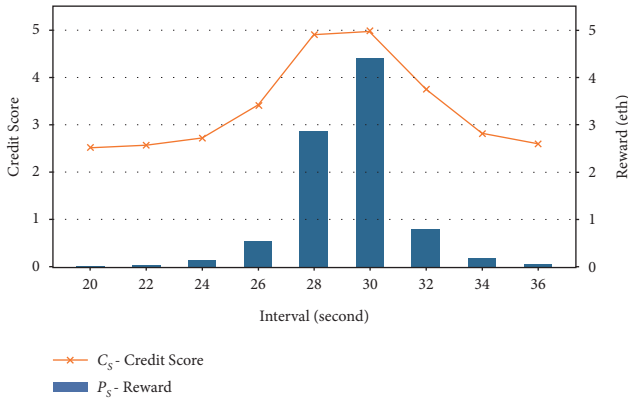


FIGURE 5: Reward changes based on different uploading periods.

We can conclude that the total credit score changes dynamically according to the occurrence of behaviors. When the period of upload is normal, the behavior will be considered as normal. Normal behavior score will be evaluated timely according to the accuracy of the normal behavior. The higher the accuracy is, the higher the score is. However, abnormal behaviors with a short period (e.g., 10 s) will change the abnormal behavior score, which will decrease the total credit score. The abnormal behavior of sensing device is not rewarded and affects its total credit score. Additionally, the average total credit score will also be reduced by abnormal behaviors and lead to the change of the corresponding reward. In this way, PoC can evaluate the behaviors of sensing devices, reward honest sensing devices, punish dishonest sensing devices, and clamp down malicious sensing devices. Thus, the system can effectively manage the behaviors of sensing devices, making the trustworthy IoT.

4.3. Cost of Data Access Control Scheme. We finally evaluated the cost of data authority management scheme running on sensing device and edge server. DACS is implemented by integrating AES and ECDSA encryption algorithm, with AES as the symmetric encryption method and ECDSA as asymmetric encryption method, and its flowchart has introduced in Section 2.3. We used secp256k1 developed by National Institute of Standards and Technology (NIST) as the elliptic curve required by ECDSA. Specifically, the prime order of the elliptic curve is set as $p = 2^{256} - 2^{32} - 2^9 - 2^8 - 2^7 - 2^6 - 2^4 - 1$, and hash function adopts SHA256 with ECDSA defined by ANSI X9.62 and finally realizes this algorithm in Java with Signature Class. AES algorithm adopts AES-128, which uses 10 rounds for 128-bit keys. The system clock tick of sensing device is set to 0.5 ms, and we collected the timestamp of each composition of DACS. The results are average values calculated from multiple experiments.

ECDSA is a common algorithm utilized for signature in blockchain system. DACS integrate AES with ECDSA, which increases the cost of data uploading of sensing devices to some extent. Figure 6 shows the impact of DACS on upload efficiency of sensing device. The efficiency of ECDSA is low and takes up most running time of DACS, especially

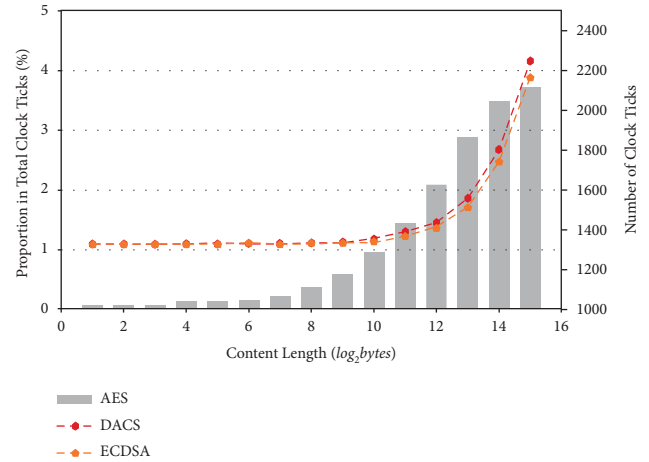


FIGURE 6: Impact of DACS on upload efficiency.

obvious when the content length is small. On the contrary, the efficiency of AES is very high, at a minimum of 1.2% of the total clock ticks. In smart city, IoT devices generally upload small batch data. When encrypting these data (less than 2^{13} bytes), AES accounts for less than 2% of total time cost. We can conclude that DACS sacrifices very little cost of time but brings high security and access control functionality.

We fixed the content length to 2^6 bytes and calculate the average total time cost of DACS, as shown in Figure 7. DACS consists of four step: production of symmetric key, symmetric encryption based on AES, signature, and decryption based on ECDSA. The first three steps are completed by sensing device, and the last step is done by edge server. The efficiency of AES based symmetric key generation and encryption is very high, only taking 2 ms and 14 ms. The efficiency of ECDSA in sensing device is relatively low, taking 667 ms, and the result is not ideal, while the decryption of ECDSA in the edge server is very fast, and it only takes 9 ms. We can conclude that the time cost of DACS is within acceptable range. However, the time cost of DACS in sensing devices is high, which can be significantly improved.

DACS effectively and securely realizes the privacy protection and trusted sharing of data, but its sharing process is complicated. “One-to-one” data sharing scheme is not the best solution for high concurrent access control requirements. Therefore, in the future, we will consider introducing CP-ABE in blockchain and improving it to a specific access scheme for IoT [24]. CP-ABE relies on an access structure for encryption, and all users who satisfy the access structure can decrypt and get the plaintext message. This one-time encryption realizes the access control of multiple users, which is a feasible and promising solution in future work.

4.4. Performance Proof-of-Contribution. Getting the performance of PoC means we need to prove the availability of PoC mechanism towards IoT devices in reality. In addition, an appropriate consensus algorithm should be chosen as the underlying consensus protocol in this subsection, which can

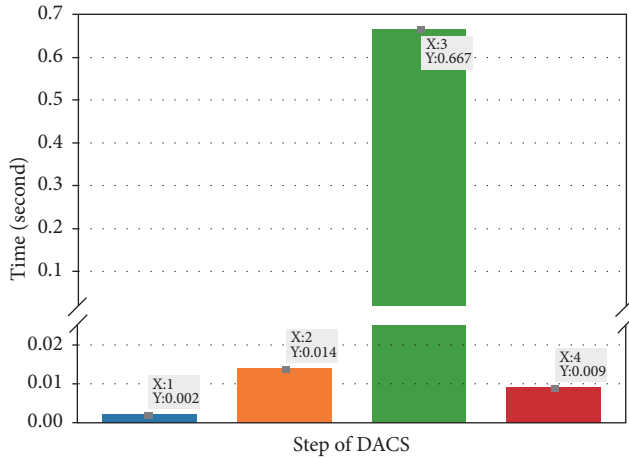


FIGURE 7: Total time cost of DACS. The four steps, respectively, represent production of symmetric key, symmetric encryption based on AES, signature, and decryption based on ECDSA.

be combined with PoC to serve IoT devices. Concretely, we deployed Ethereum and Hyperledger fabric on three cloud servers and configured Hyperledger Caliper [25] on one of the clouds, which is an effective performance monitoring tool for blockchain. Each cloud server will act as an edge server accessing the same number of IoT devices, and Caliper will monitor the throughput performance of PoC mechanism in blockchain systems with different underlying consensus protocols. Considering the large-scale IoT devices in practical application scenarios, we built a consortium chain in Ethereum, as in Section 3.1. This consortium chain is based on PoW consensus algorithm, and we set an adaptive mining mechanism similar to the public chain, which has the ability to adjust the difficulty of mining timely and maintain the stability of the system. Besides PoW, we think that PBFT algorithm with excellent fault tolerance is also one of the available underlying consensus. However, it should be noted that we chose an earlier version of fabric to run the PBFT algorithm normally because of being not well implemented.

We set the frequency of IoT devices interacting with blockchain network to two requests/s and continuously increased the number of IoT devices loaded by each edge server. Figure 8 is the comparison of throughput performance between PBFT and PoC, which is plotted from the average value of 10 rounds of experiments. PBFT shows excellent performance when only a few IoT devices join the network, but it is a consensus serving the consortium chain after all. Since the communication complexity of PBFT is $O(n^2)$, a large number of IoT devices accessing the network will significantly increase the workload of message broadcasting process. Concurrently, the number of consensus nodes, server hardware, and other factors also limit the scalability of PBFT. This shortcoming is also reflected in the figure, as the number of IoT devices connected to each edge server increases continuously, the throughput of PBFT has a dramatic decrease, and eventually fails to function properly. Comparatively, although proof-of-contribution mechanism based on PoW does not have superior performance, it has a remarkable stability. PoC dynamically adjusts the difficulty of mining, and it leads to

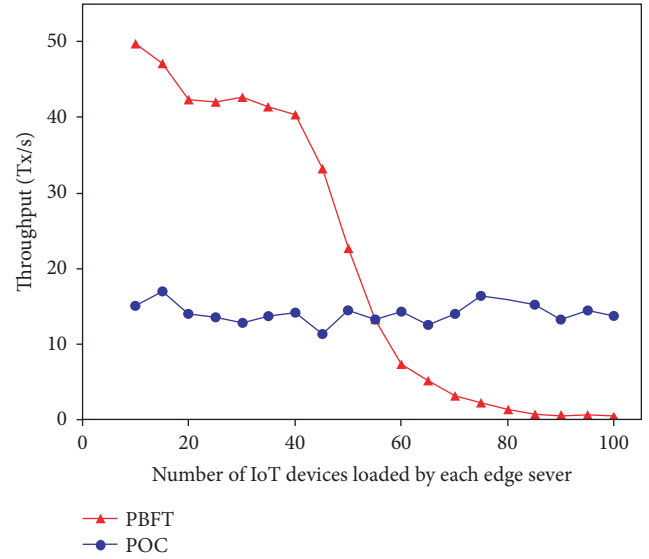


FIGURE 8: Comparison of throughput and load capacity between PoC based on PoW and PBFT.

throughput performance of blockchain always maintaining at a consistent level. It also means that PoC has more practical and stable throughput when a large number of IoT devices participate in the network. The stability of PoW is what PBFT cannot do for the time being, which is the reason for choosing PoC based on PoW finally.

5. Conclusion

A blockchain-edge computing hybrid system is presented to provide trustworthy IoT services in smart cities. A novel proof-of-contribution consensus mechanism is proposed to regulate the behavior of nodes, especially IoT device nodes. PoC can detect and prevent abnormal behaviors realized by modifying the data upload frequency, such as greed, absenteeism, or sabotage, so as to prevent them from damaging system. A data access control scheme is proposed to secure the data authenticity, especially to protect the private data of IoT devices. We implemented the concrete system on Ethereum platform. The extensive evaluations and analyses show that the proposed PoC mechanism can effectively manage behaviors of nodes in the system, securing the system from attacks, and the cost of the designed data access control scheme is within reasonable range. However, Ethereum-based BEHS is an application prototype system, and there are still some shortcomings that need to be continuously updated and improved, such as low concurrency caused by smart contract and limited functionality of sensing device. In the future work, we will continue to explore the implementation approaches of the system, such as architecture with pluggable consensus algorithm and cross-chain communication to improve the expansibility and purity of the system.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was supported by the National Science Foundation of China (61702354 and 61876121), Scientific Research Project of Suzhou University of Science and Technology (XKZ2017004), Key Laboratory of Mobile Interconnection Technology Engineering of Jiangsu Province (JSWLW2017004), Graduate Research Innovation Project (SKSJ18_012 and SJCX19_0963), and Educational Reform Project of Suzhou University of Science and Technology (SKJG18_05).

References

- [1] X. Wang, X. Chen, Z. Li, and Y. Chen, "Access delay analysis and optimization of NB-IoT based on stochastic network calculus," in *Proceedings of the 2018 IEEE International Conference on Smart Internet of Things (SmartIoT)*, pp. 23–28, IEEE, Xi'an, China, August 2018.
- [2] T. Wang, H. Ke, X. Zheng, K. Wang, A. K. Sangaiah, and A. Liu, "Big data cleaning based on mobile edge computing in industrial sensor-cloud," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 2, pp. 1321–1329, 2019.
- [3] L. U. Khan, I. Yaqoob, N. H. Tran, S. M. Ahsan Kazmi, T. N. Dang, and C. S. Hon, "Edge-computing-Enabled smart cities: a comprehensive survey," *IEEE Internet of Things Journal*, vol. 7, no. 10, pp. 10200–10232, 2020.
- [4] V. A. F. Almerida, D. Doneda, and D. S. A. Jacqueline, "Cyberwarfare and digital governance," *IEEE Internet Computing*, vol. 21, no. 2, pp. 68–71, 2017.
- [5] A. Reyna, C. Martín, J. Chen, E. Soler, and M. Díaz, "On blockchain and its integration with iot. challenges and opportunities," *Future Generation Computer Systems*, vol. 88, pp. 173–190, 2018.
- [6] S. Nakamoto, "Bitcoin: a peer-to-peer electronic cash system," *Decentralized Business Review*, Article ID 21260, 2008.
- [7] H. Yu, Z. Yang, and R. O. Sinnott, "Decentralized big data auditing for smart city environments leveraging blockchain technology," *IEEE Access*, vol. 7, pp. 6288–6296, 2018.
- [8] G. Yu, X. Zha, X. Wang et al., "Enabling attribute revocation for fine-grained access control in blockchain-iot systems," *IEEE Transactions on Engineering Management*, vol. 67, no. 4, pp. 1213–1230, 2020.
- [9] P. Kumar, R. Kumar, G. P. Gupta, and R. Thirupathi, "A Distributed framework for detecting dos attacks in smart contractbased blockchainiot systems by leveraging fog computing," *Transactions on Emerging Telecommunications Technologies*, vol. 32, no. 3, pp. 1–31, 2020.
- [10] Y. Zhang, S. Kasahara, Y. Shen, X. Jing, and J. Wan, "Smart contract-based access control for the internet of things," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 1594–1605, 2018.
- [11] J. Huang, L. Kong, and G. Chen, "Towards secure industrial iot: blockchain system with credit-based consensus mechanism," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 6, pp. 3680–3689, 2019.
- [12] J. Pan, J. Wang, A. Hester, I. Alqerm, Y. Liu, and Y. Zhao, "Edgechain: an edge-iot framework and prototype based on blockchain and smart contracts," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 4719–4732, 2018.
- [13] R. Chen, I. P. Tu, K. E. Chuang, Q.-X. Lin, S.-W. Liao, and W. Liao, "Endex: Degree of mining power decentralization for proof-of-work based blockchain systems," *IEEE Network*, vol. 34, no. 6, pp. 266–271, 2020.
- [14] F. Saleh, *Blockchain without Waste: Proof-Of-Stake*, Social Science Electronic Publishing, Rochester, NY, USA, 2018.
- [15] S. Biswas, K. Sharif, F. Li, S. Maharjan, S. P. Mohanty, and Y. Wang, "Pobt: a light weight consensus algorithm for scalable iot business blockchain," *IEEE Internet of Things Journal*, vol. 7, no. 3, pp. 2343–2355, 2019.
- [16] W. Viriyasitavat, L. D. Xu, and Z. Bi, "Managing qos of internet-of-things services using blockchain," *IEEE Transactions on Computational Social Systems*, vol. 6, no. 6, pp. 1357–1368, 2019.
- [17] S. Guo, Y. Dai, S. Guo, X. Qiu, and F. Qi, "Blockchain meets edge computing stackelberg game and double auction based task offloading for mobile blockchain," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 5, pp. 5549–5561, 2020.
- [18] E. K. Wang, Z. Liang, C. M. Chen, S. Kumari, and M. K. Khan, "PoR. X.: A reputation incentive scheme for blockchain consensus of IIoT," *Future Generation Computer Systems*, vol. 102, pp. 140–151, 2020.
- [19] H. Song, N. Zhu, R. Xue, J. He, K. Zhang, and J. Wang, "Proof-of-Contribution consensus mechanism for blockchain and its application in intellectual property protection," *Information Processing & Management*, vol. 58, no. 3, Article ID 102507.
- [20] C. Decker and R. Wattenhofer, "Information propagation in the bitcoin network," in *Proceedings of the IEEE Thirteenth International Conference on Peer-To-Peer Computing*, Trento, Italy, April 2013.
- [21] N. T. Courtois, M. Grajek, and R. Naik, *Optimizing Sha256 in Bitcoin Mining*, Springer, New York, NY, USA, 2014.
- [22] A. I. Abhi and S. Y. Shin, "Bus: a blockchain-enabled data acquisition scheme with the assistance of uav swarm in internet of things," *IEEE Access*, vol. 7, no. 1, pp. 103231–103249, 2019.
- [23] <https://github.com/ethereum/go-ethereum>. 2020.
- [24] K. P. Yu, L. Tan, M. Aloqaily, H. Yang, and Y. Jararweh, "Blockchain-enhanced data sharing with traceable and direct revocation in IIoT," *IEEE transactions on industrial informatics*, vol. 17, no. 11, pp. 7669–7678, 2021.
- [25] T. Wang, J. Guo, and S. Ai, "RBT: a distributed reputation system for blockchain-based peer-to-peer energy trading with fairness consideration," *Applied Energy*, vol. 295, Article ID 117056, 2021.

Research Article

Security Analysis of Intelligent System Based on Edge Computing

Yibo Han,¹ Weiwei Zhang,² and Zheng Zhang ³

¹Nanyang Institute of Big Data Research, Nanyang Institute of Technology, Nanyang, Henan 473000, China

²Nanyang Fangyuan Limited Liability Accountant Firm, Nanyang, Henan 473000, China

³School of Computer and Software, Nanyang Institute of Technology, Nanyang, Henan 473000, China

Correspondence should be addressed to Zheng Zhang; zhangzheng@nyist.edu.cn

Received 1 June 2021; Revised 1 July 2021; Accepted 31 July 2021; Published 18 August 2021

Academic Editor: Shahram Babaie

Copyright © 2021 Yibo Han et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

At present, artificial intelligence technology is widely used in society, and various intelligent systems emerge as the times require. Due to the uniqueness of biometrics, most intelligent systems use biometric-based recognition technology, among which face recognition is the most widely used. To improve the security of intelligent system, this paper proposes a face authentication system based on edge computing and innovatively extracts the features of face image by convolution neural network, verifies the face by cosine similarity, and introduces a user privacy protection scheme based on secure nearest neighbor algorithm and secret sharing homomorphism technology. The results show that when the threshold is 0.51, the correct rate of face verification reaches 92.46%, which is far higher than the recognition strength of human eyes. In face recognition time consumption and recognition accuracy, the encryption scheme is basically consistent with the recognition time consumption in plaintext state. It can be seen that the security of the intelligent system with this scheme can be significantly improved. This research provides a certain reference value for the research on the ways to improve the security of intelligent system.

1. Introduction

With the rapid development of mobile network, multimedia data on network edge devices are increasing rapidly. The network communication load and storage space of the traditional cloud computing intelligent system are impacted. With the improvement of the real-time requirements of the network, the edge computing arises at the historic moment [1]. Relevant research shows that as of October 30, 2020, 50% of multimedia data have been preprocessed, forwarded, stored, and other operations through the Internet edge [2, 3]. The cloud computing mode of centralized processing will fall into the demand of real-time and privacy protection that cannot complete the common processing of all programs, and edge computing has become a new direction of development [4]. Face recognition has the advantages of incompatibility, mobility, uniqueness, directness, and friendliness and has become the mainstream technology for user authentication in intelligent systems [5]. Face recognition technology mainly distinguishes different faces through the distinguishability of faces. Due to the openness

of the Internet environment, the authentication system based on biometrics has a great risk of privacy leakage [6]. To improve the security of user identity authentication in intelligent system, an identity authentication scheme based on edge computing is proposed. The original face image is processed by convolution neural network, and the feature vector of face is extracted. The user identity registration technology based on secure nearest neighbor algorithm and the user identity authentication technology based on secret sharing homomorphism are introduced.

With the development of industrial Internet of things, the type and number of industrial equipment increase. Through established a noninvasive load monitoring system through recurrent neural network long-term memory and identified the power equipment through edge calculation. The research results show that the average random recognition rate of the system can reach 88% [6]. The mobile Internet of things can process a large amount of real-time data. To alleviate the contradiction between the resource constraints of mobile devices and the requirements of users to reduce processing delay and extend battery life, Huang

et al. and other scholars proposed a computing offload method for cloud edge computing supporting the Internet of things and solved the multiobjective optimization problem of task offload in cloud computing through nondominant sorting genetic algorithm III [7]. Researchers proposed that mobile edge computing and UAV base station have become a promising technology in the Internet of things and designed an online edge processing scheduling algorithm based on Lyapunov optimization. When the data rate is low, it tends to reduce the frequency of edge processor. When the data rate is high, it will flexibly allocate bandwidth for edge data unloading [8]. After investigating the development of artificial intelligence, edge computing, and the occurrence of big data, scientific team believe that when people extract intelligent information from Internet of things nodes, the user's information data are vulnerable to network attacks and information leakage, that is, the data richness and data analysis of intelligent management system form a great risk of infringement on the user's privacy [9]. With the development of intelligent transportation system, video analysis technology has become a potential technology to improve vehicle network security, but a large number of video data transmission brings great pressure to vehicle network. A video analysis framework is proposed, which integrates multiaccess edge computing and block chain technology into the Internet of things to optimize the transaction throughput of block chain system [10]. Researchers proposed a vehicle edge planner based on two-stage machine learning, to provide better driving service for drivers [11].

Face recognition is the main way for most intelligent systems to identify users, especially for intelligent monitoring systems. When the distance between monitoring and face is too far, the success and accuracy of capturing face are reduced. Therefore, Scholars use deep convolution neural network to improve the resolution of captured image and complete face feature extraction and classification [12]. In image recognition, the Science team applied hierarchical clustering technology to divide the database into some interrelated clusters and sort them and then compared the classification effect through deep convolution neural network [13]. A lightweight convolutional neural network structure is proposed, which uses smaller filter size and depth separable convolution to improve the nonlinear performance of the model and complete the mapping from the original low-resolution image to the high-resolution image [14]. Other researchers have successfully extracted the host's watermark image under various attacks by using the nonembedded blind image watermarking algorithm based on mapping residual convolution neural network [15]. Modern team proposed a distributed storage computing k-nearest neighbor algorithm for data processing in the Internet of things. By performing distributed computing on each storage node, the algorithm effectively performs k-nearest neighbor search and improves the speed of data processing [16]. Scholars have proposed an automatic license plate image recognition technology, which uses the boundary tracking method to segment the contour, and then uses the nearest neighbor algorithm to complete the image

recognition, which has high security [17]. To solve the problem of encrypted traffic identification, some scholars proposed an encrypted network behavior identification method based on dynamic time warping and k-nearest neighbor [18].

To sum up, a lot of research has been carried out in edge computing, secure nearest neighbor algorithm, face recognition, intelligent system, user data privacy protection, and so on. However, in the aspect of improving the security of intelligent system, there is still a lack of research on using edge computing, convolutional neural network face feature extraction, and secure nearest neighbor algorithm to improve the security of face recognition. In view of this, this paper proposes an intelligent system security enhancement scheme based on edge computing, which uses convolution neural network to extract the feature vector of face image and uses secure nearest neighbor algorithm to protect the user privacy.

2. Research on Security Enhancement Technology of Intelligent System Based on Face Recognition

2.1. Face Feature Vector Extraction Based on CNN. Edge in edge computing refers to network devices with data storage capacity and data computing capacity, which are distributed between terminal data source and cloud server [19]. Edge computing is both the data owner and the data user, which also means that the data requests between cloud computing center and edge computing devices are bidirectional requests [20, 21]. At the same time, the data at the edge of edge computing is divided into uplink and downlink. Uplink refers to cloud computing services, and downlink refers to Internet of things services. While sending and receiving data to the cloud service center, edge computing also takes into account part of the data computing and storage tasks of the cloud Computing Center. See Figure 1 for details.

Due to the uniqueness, incompatibility, direct friendliness, and other characteristics of face recognition, it has become an authentication method in a variety of intelligent systems, and its security directly determines the security of intelligent systems. Therefore, this paper proposes a privacy protection technology in an intelligent face authentication system based on edge computing [22]. The main technologies of face recognition include face detection, face data preprocessing, face feature extraction, similarity measurement, and discriminant classification, and finally output the recognition results [23]. In this study, convolutional neural network (CNN) is used to assist in face authentication of intelligent system. Through learning a large number of face data, the face information is digitally represented to form a deep CNN model for face feature extraction. The basic structure of CNN includes convolution layer, pooled sampling layer, and full connection layer; see Figure 2 for details.

In convolution layer, convolution core is used to traverse the image, and the corresponding data in the same region of the image are accumulated to activate function operation as the output of a single neuron.

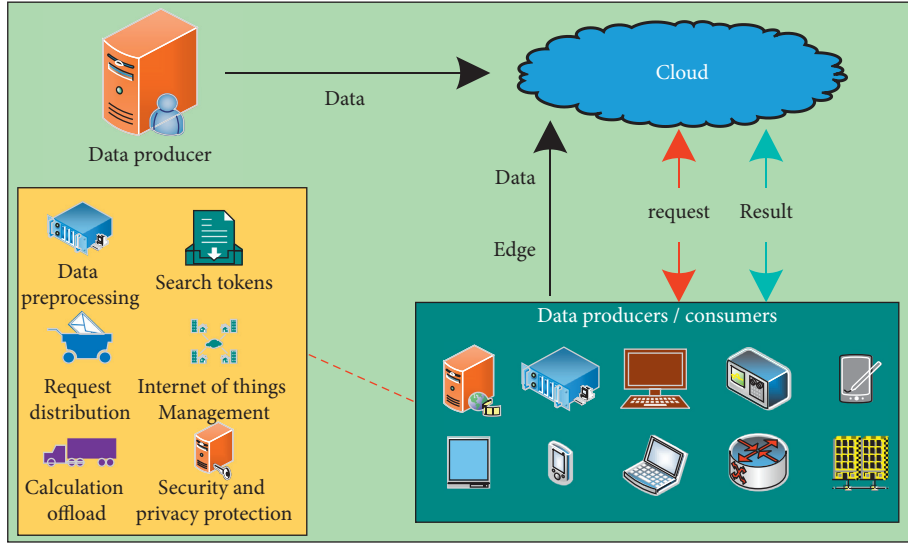


FIGURE 1: Edge computing model.

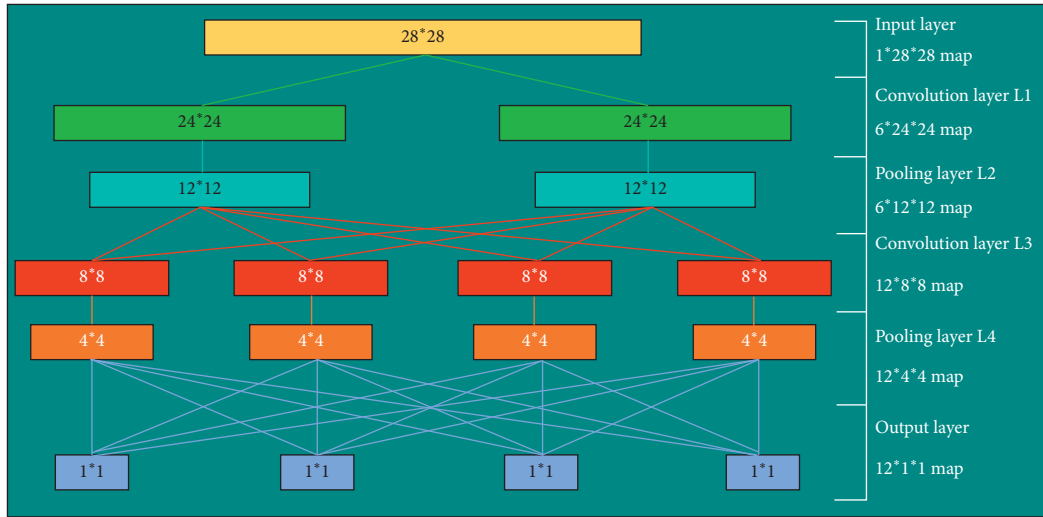


FIGURE 2: Basic structure of convolutional neural network.

$$x_j^l = F\left(\sum_{i,j} \omega_{ij}^l \otimes x_i^{l-1} + b_j^l\right). \quad (1)$$

In formula (1), x_j^l refers to the j characteristic graph on the l layer of CNN; $F(\cdot)$ is the activation function in the network; and ω_{ij}^l and b_j^l refer to the weight parameter and bias parameter in turn.

$$H_\theta(x) = \begin{bmatrix} p(y^i = 1|x^i; \theta) \\ p(y^i = 2|x^i; \theta) \\ \dots \\ p(y^i = m|x^i; \theta) \end{bmatrix} = \frac{1}{\sum_{j=1}^m e^{\theta_j^T x^i}} \begin{bmatrix} e^{\theta_1^T x^i} \\ e^{\theta_2^T x^i} \\ \dots \\ e^{\theta_m^T x^i} \end{bmatrix}. \quad (2)$$

In formula (2), x^i is the input vector of the classifier; y^i is the sample category, $y^i \in \{1, 2, \dots, m\}$ is the sample

category, and m is the total number of samples, so $p(y^i = m|x^i; \theta)$ is the probability estimate.

$$J(\theta) = -\frac{1}{N} \left[\sum_{i=1}^N \sum_{j=1}^m 1\{y^i = j\} \log \frac{e^{\theta_j^T x^i}}{\sum_{j=1}^m e^{\theta_j^T x^i}} \right]. \quad (3)$$

Equation (3) is the objective loss function of softmax classifier, where the meaning of each letter is the same as above. In the research process, the cosine similarity function is used to verify whether the two feature vectors belong to different face images of the same person, as shown in equation (4).

$$\text{COS}(f_1, f_2) = \frac{f_1^T f_2}{\|f_1\| \|f_2\|}. \quad (4)$$

In equation (4), f_1 and f_2 are all arbitrary face feature vectors, where $f_1 = (a_1, a_2, \dots, a_m)$ and

$f_2 = (b_1, b_2, \dots, b_m)$ obey Gaussian distribution of 0-means. Whether two eigenvectors belong to the same person or not is measured by calculating the similarity of two eigenvectors in multidimensional space. In the process of research, Shamir threshold scheme is selected to protect sensitive data. The secret information is recorded as s , divided into n parts, and distributed to n users. A perfect (t, n) secret sharing threshold requires at least T Information holders to reconstruct the secret information.

$$f(x) = \sum_{i=1}^t f_i(x) = \sum_{i=1}^t y_i \prod_{j=1, j \neq i}^t \frac{x - x_j}{x_i - x_j}. \quad (5)$$

Equation (5) shows the process of secret information reconstruction by t information cooperators, and $(x_i, x_j) (1 \leq i \leq t)$ is the subkey owned by t information holders; x_i is a nonzero constant, which is open to all information holders; and y_i is the unique subkey of a single information holder.

Figure 3 shows the CNN structure responsible for face feature extraction, which consists of four convolution layers and maximum pooling to recognize face features hierarchically; The output of one-dimensional feature is realized by a fully connected layer; The softmax output layer is used to output feature categories.

Figure 4 shows the specific model parameters of convolutional neural network used in the research process. Totally, 2800 categories are selected as the training data, that is, the final output size of softmax output layer is 2800. It can be seen that with the extension of network structure, the dimension of feature graph is decreasing, and it becomes a highly abstract feature vector in the last hidden layer.

$$y^{j(r)} = F\left(b^{j(r)} + \sum_i k^{ij(r)} * x^{i(r)}\right). \quad (6)$$

In equation (6), x^i refers to the feature map of the input of layer i ; y^j refers to the feature map output by the j layer; k^{ij} is the convolution kernel between x^i and y^j ; “*” calculate the symbol for convolution; b^j is the configuration parameter corresponding to the characteristic graph of the j th output layer; and r is the weight sharing area.

$$F(x) = \begin{cases} ax, & x < 0, \\ x, & x \geq 0. \end{cases} \quad (7)$$

Equation (7) is the parametric relu activation function of activated neurons, where a is the parameter involved in training.

$$y_{j,k}^i = \max_{0 \leq m, n < s} \{x_{j-s+m, k-s+n}^i\}. \quad (8)$$

Formula (8) is the maximum pooling formula, y^i is the i th output characteristic graph, in which each neuron comes from the nonoverlapping region with the size of $s \times s$ in x^i .

$$y_j = F\left(\sum_i x_i^1 \cdot \omega_{i,j}^1 + \sum_i x_i^2 \cdot \omega_{i,j}^2 + b_j\right). \quad (9)$$

Formula (9) is the calculation formula of the neurons in the last hidden layer. The corresponding neurons in the last

convolution layer are expressed as x^1 and x^2 , the weight parameters are expressed as ω^1 and ω^2 , the bias parameter is b , and the activation function is $F(\cdot)$.

$$y_i = \frac{\exp(y_i')}{\sum_{k=1}^n \exp(y_k')}, \quad (10)$$

$$y_k' = \sum_{i=1}^1 60x_i \cdot \omega_{i,k} + b_k. \quad (11)$$

Equation (10) is responsible for predicting the probability distribution of n categories. In equation (11), the calculation result of 160-dimensional eigenvector is used as the input of category k , and the output is y_k . The bias parameter of class k is b_k . The input of the i layer is characterized by x_i ; $\omega_{i,k}$ are the weights corresponding to the features of class k and layer i .

2.2. Privacy Protection Scheme for Intelligent System.

After extracting face feature data through CNN, privacy protection scheme should be set to protect face data stored in the location of edge computing node [24]. When users register their identity through an edge computing node, a privacy protection scheme based on the nearest security neighbor is set.

As shown in Figure 5, when the user registers, the camera collects face data and uploads it to the edge computing node. The authority allocation agency is responsible for transmitting the corresponding encrypted authority vector to the edge computing node, and the edge computing node extracts face features and encrypts them [25]. In this process, there is a 160-dimensional random bit vector s and two 160×160 random invertible matrices M_1 and M_2 . The key is shared by all n edge computing nodes.

$$f_i = (f_{i,1}, f_{i,2}, \dots, f_{i,160})^T. \quad (12)$$

Formula (12) is the expression of face feature vector of registered user f_i , where T is the threshold value of face verification, i is the output feature map, and the edge computing node transforms formulae (12) into (13).

$$\hat{f}_i = \left(\frac{f_{i,1}}{\|f_i\|}, \frac{f_{i,2}}{\|f_i\|}, \dots, \frac{f_{i,160}}{\|f_i\|} \right)^T. \quad (13)$$

In equation (13), $\|f_i\|$ refers to the 2- norm of the face feature vector $f_i = (f_{i,1}, f_{i,2}, \dots, f_{i,160})^T$.

$$\begin{cases} \hat{f}_{ia}[j] = \hat{f}_{ib}[j] = \hat{f}_i[j], & \text{if } S[j] = 0, \\ \hat{f}_{ia}[j] + \hat{f}_{ib}[j] = \hat{f}_i[j], & \text{if } S[j] = 1. \end{cases} \quad (14)$$

In equation (14), $j \in [1, 160]$, when $[j] = 0$, there is $\hat{f}_{ia}[j] = \hat{f}_{ib}[j] = \hat{f}_i[j]$, When $S[j] = 1$, $\hat{f}_{ia}[j]$ is an arbitrary real number and $\hat{f}_{ia}[j] + \hat{f}_{ib}[j] = \hat{f}_i[j]$ exists. Where S is the encryption key, the vector $(\hat{f}_{ia}, \hat{f}_{ib})$ can be obtained by substituting \hat{f}_i and S into equation (14). Combined with the encryption key M_1, M_2 , $(M_1^T \hat{f}_{ia}, M_2^T \hat{f}_{ib})$ can be obtained as the feature vector for encryption in the privacy protection scheme. The fluorite protection scheme based on the secure nearest neighbor algorithm makes a lightweight encryption

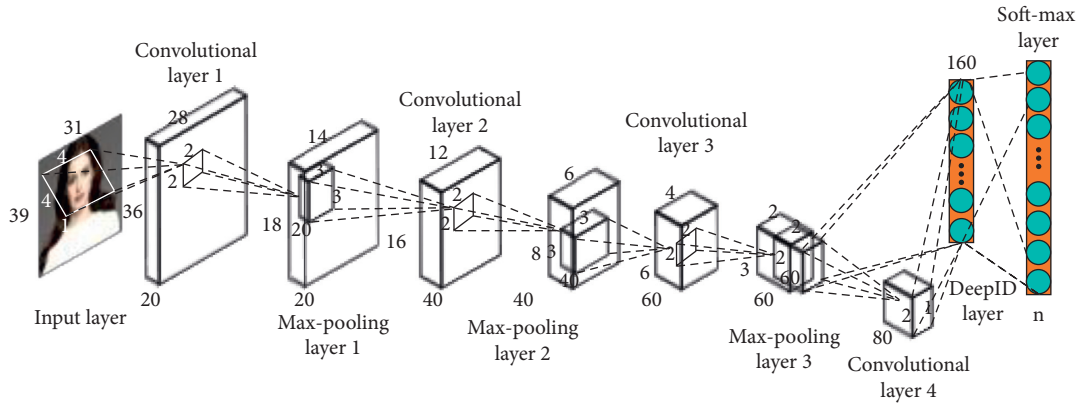


FIGURE 3: CNN structure for face feature extraction.

Network level	Passageway	Filter size	Step	Enter the size	Output size
Convolution layer 1	20	4*4	1	39x31x1	36x28x20
Pooling layer 1	20	2*2	2	36x28x20	18x14x20
Convolution layer 2	40	3*3	1	18x14x20	16x12x40
Pooling layer 2	40	2*2	2	16x12x40	8x6x40
Convolution layer 3	60	3*3	1	8x6x40	6x4x60
Pooling layer 3	60	2*2	2	6x4x60	3x2x60
Convolution layer 4	80	2*2	1	3x2x60	2x1x80
Fully connected layer	1	-	-	2x1x80	160x1
Softmax layer	1	-	-	160x1	2800

FIGURE 4: Parameters of convolution neural network model.

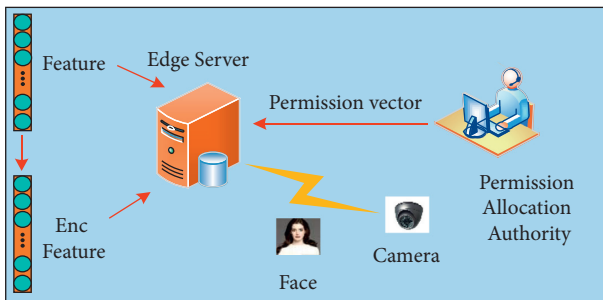


FIGURE 5: Identity registration process of privacy protection scheme based on secure nearest neighbor algorithm.

of face feature vectors and stores the local database with edge computing equipment. Users can obtain the corresponding information access rights after they pass the identity authentication, so as to realize the privacy protection of users.

When an edge computing node is requested to perform identity authentication, the node randomly selects $(t-1)$ devices, which come from other edge computing. The two devices cooperate through secret sharing homomorphism technology and aggregate the obtained calculation results through cloud computing center to complete the acquisition of user permission information [25]. The details are shown in Figure 6.

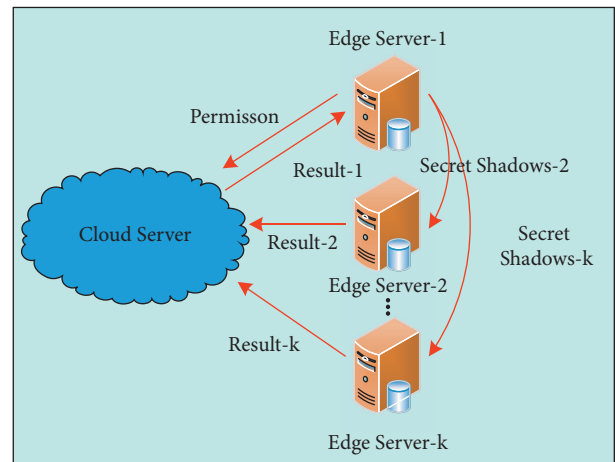


FIGURE 6: Identity authentication process of privacy preserving scheme based on secret sharing homomorphism.

After a series of preprocessing, such as redundant data clipping, interference noise filtering, image scaling, and so on, a 160-dimensional feature vector $f_q = (f_{q,1}, f_{q,2}, \dots, f_{q,160})^T$ is proposed from the image through CNN model, in which f_q is the face feature vector of the authenticated user, and T is the threshold value of face verification.

$$Q[j] = \begin{cases} -1, & f_{i,j} < 0, \\ 1, & f_{i,j} \geq 0. \end{cases} \quad (15)$$

In equation (15), Q is the user requesting authentication, j is the dimension, and $j \in [1, 160]$ and f_i are the face feature vectors of registered users.

$$U_j^i[k] = (\tilde{f}_{qj}[k] + \tilde{f}_{ij}[k]) \prod_{l=1, l \neq j}^t \frac{-x_l}{x_j - x_l} \pmod{p}. \quad (16)$$

Equation (16) is the expression of the intermediate vector U_j^i , where $k \in [1, 160]$, $(\tilde{f}_{q1}[k], \tilde{f}_{q2}[k], \dots, \tilde{f}_{qj}[k])$ is the t sub secret of the eigenvector $\tilde{f}_i[k]$, t is the threshold value in secret sharing homomorphism, and p is a large prime number greater than n . t edge computing encrypts U_j^i and sends it to the cloud server. The cloud server summarizes all the information and compares the cosine similarity between the eigenvector f_q and the eigenvector f_i through equation (17). Cosine similarity can calculate the similarity between any two feature vectors in multidimensional space and measure the similarity mainly by the angle. According to the definition of cosine similarity, the cosine values of the angles between all matching vectors and reference vectors are similar. When using cosine similarity as a constraint condition for face recognition, it can effectively reduce the false matching points.

$$\text{COS}(f_q, f_i) = \sum_{k=1}^{160} R_i[k] Q[k] 2^{\sum_{j=1}^n U_j^i[k]}, \quad (17)$$

where R is the symbol vector of registered users, Q is the symbol vector of authenticated users, f_q and f_i are the feature vectors of human face, and $\text{COS}(\cdot)$ is the calculation formula of cosine similarity, $k \in [1, 160]$.

3. Analysis of Security Effect of Intelligent System

3.1. Training Effect of Convolution Neural Network. CASIA Webface data set is selected as the training set of convolutional neural network. The data set contains more than 10000 categories of data, a total of ab better. It can be seen that when the false-positive rate (FPR) is the same, the true rate (TPR) of CNN model is always higher than that of ANN moing set. After the research process, LFW face data set is selected as the verification set of CNN model. There are 5749 categories of objects in the data set, including 13233 face images, of which 1680 objects have two or more face images. The maximum number of iterations of the network is 240000, the test interval is 2000, the number of iterations to complete a test is 129, and the learning rate is 0.001. Every 40000 iterations of the network, 0.1 is used as an index to update the learning rate, and the network is trained in CPU mode [26].

Figure 7 shows that with the increase of the number of iterations, the test loss value in the network training process decreases gradually. When the number of iterations is 50000, the loss value decreases to the minimum, and then gradually becomes stable. In the process of network training, the

model test accuracy increases with the increase of the number of iterations. When the number of iterations is 50000, the test accuracy reaches the maximum, and then gradually becomes stable, and the convolutional neural network training is successful. The LFW data set is selected as the validation set of the convolutional neural network model after training, and 6000 pairs of face images are selected. In total, 3000 pairs of face data in these images are positive examples, marked as 1, and the remaining images are from different objects and are marked as 0. The trained convolution neural network is used to extract the feature vectors of 6000 pairs of faces in the data set. According to the specific situation of the feature vectors, the cosine similarity between the feature vectors is calculated and normalized to the $[0, 1]$ interval. Different thresholds between 0.2 and 0.8 are selected to calculate the accuracy of 6000 pairs of face verification under different thresholds.

As can be seen from Figure 8, with the increase of the threshold value from 0.2 to 0.8, the accuracy rate of face verification first increases and then decreases. When the threshold value is 0.51, the accuracy of face verification reaches the maximum value, which is 92.46%, which also indicates that the accuracy of face verification of the intelligent system designed in this study can reach 92.46%, which is far beyond the recognition strength of human eyes, indicating that the proposed scheme can significantly increase the security of the intelligent system.

In the field of machine learning, receiver operating characteristic curve (ROC) is often used to evaluate the performance of the model. The true-positive rate (TPR) = [true case TP/(false counterexample FN + true case TP)] and the false-positive rate (FPR) = [false-positive case FP/(true counterexample TN + false-positive case FP)]. ROC curve is drawn with TPR and FPR as indicators. The larger the area under ROC curve is, the better the effect of the model is. Figure 9 shows that the model works well.

Figure 10 shows an example of data matching failure in the verification process, in which two images in each column are the same object. It can be seen that the reasons for the failure include exaggerated expression, special shooting angle, and partial occlusion of face. That is to say, when facial expression, action, and expression are in normal state, the model designed in this paper has good recognition and matching effect, that is, the model proposed in this paper has good application effect in protecting the privacy of data set.

3.2. Security Effect Analysis of Face Recognition in Intelligent System. To verify the face recognition security of the intelligent system designed in the research, the experiment selects the intelligent face recognition system with artificial neural network (ANN) as the core and the intelligent face recognition system with deep neural network (DNN) as the core and selects CASIA webface data set as the test set, The accuracy of the three systems in CASIA webface data set is compared. CASIA webface data set contains more than 10000 categories of data and about 500000 face images.

It can be seen from Figure 11(a) that the ROC curves corresponding to Ann and DNN are all included in the range

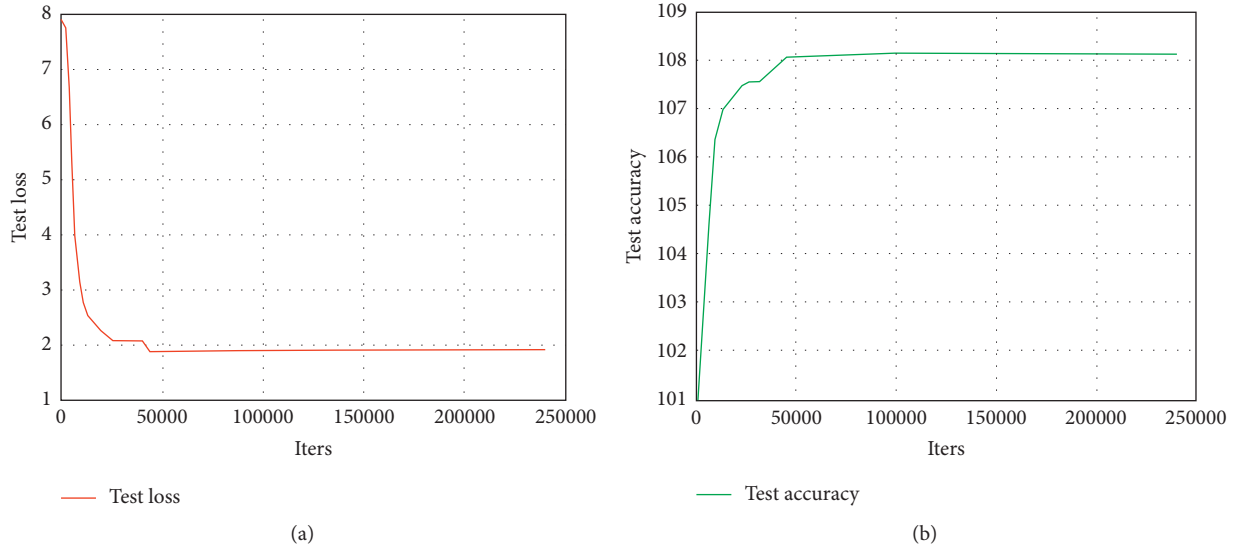


FIGURE 7: Test precision curve and test loss value curve in the process of network training. (a) Test loss value in the process of network training. (b) Test accuracy curve in the process of network training.

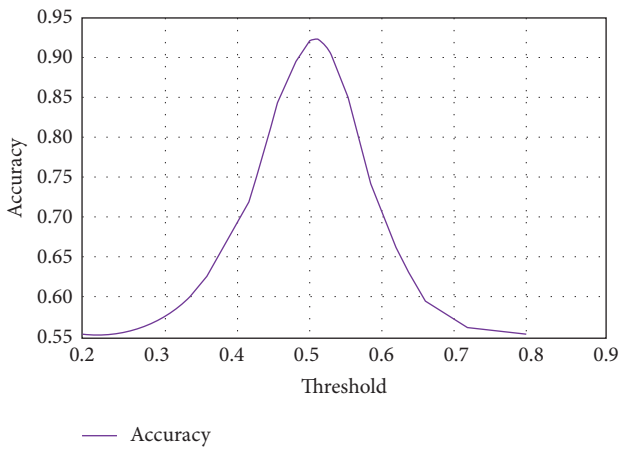


FIGURE 8: Curve of face verification accuracy with threshold.

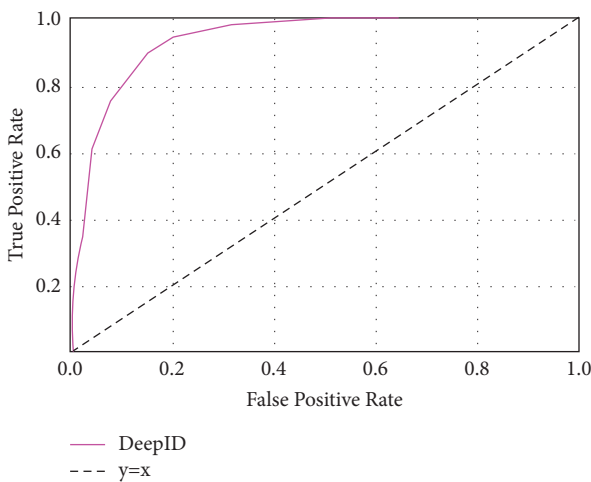


FIGURE 9: ROC curve drawn by cosine similarity of feature vector plaintext.

of the ROC curves corresponding to CNN. When the false-positive rate (FPR) is the same, the performance of the model represented by the curve with higher true rate (TPR) is better. It can be seen that when the false-positive rate (FPR) is the same, the true rate (TPR) of CNN model is always higher than that of ANN model and DNN model, and the true rate (TPR) of DNN model is always higher than that of ANN model. That is to say, the performance of CNN model is always better than ANN model and DNN model. At this time, the area under the ROC curve of ANN, DNN, and CNN is 0.8826, 0.9278, and 0.9359, respectively, which indicates that the intelligent system based on convolutional neural network designed in this paper can achieve better application effect in the process of face recognition verification. Figure 11(b) shows that the convergence speed of the intelligent system based on convolutional neural network (CNN) is faster than that based on ANN and DNN, which indicates that the former can complete the whole process faster in face recognition and verification.

As can be seen from Figure 12, the time consumption of the privacy protection scheme based on the secure nearest neighbor algorithm combined with the secret sharing homomorphism technology is mainly concentrated on the feature vector extraction, recognition, and encryption. It can be seen that the time consumption of face recognition in plaintext state is the lowest, and the time consumption of face recognition in the proposed algorithm is basically equal to that in plaintext state, which indicates that the proposed technology can quickly complete the user's identity registration and verification without too much interaction process on the premise of protecting the user's privacy and security, The role of edge fitting computing in the system also reduces the security degradation of intelligent system caused by too much interaction to a certain extent. In addition, the convolution neural network is used to extract the features of face image instead of the original face image,

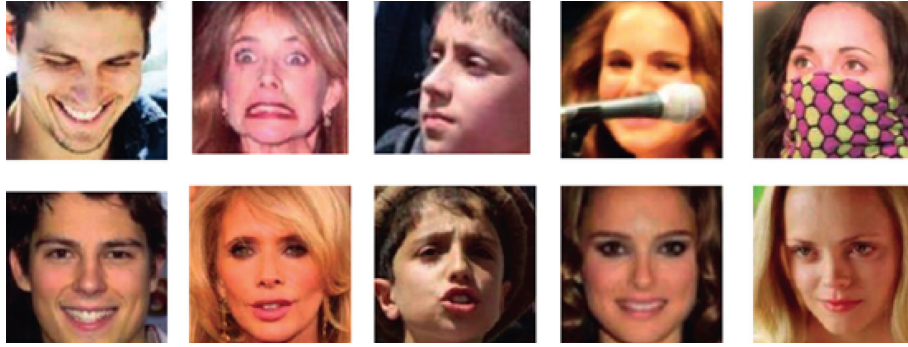


FIGURE 10: Failed data instance validation set matching.

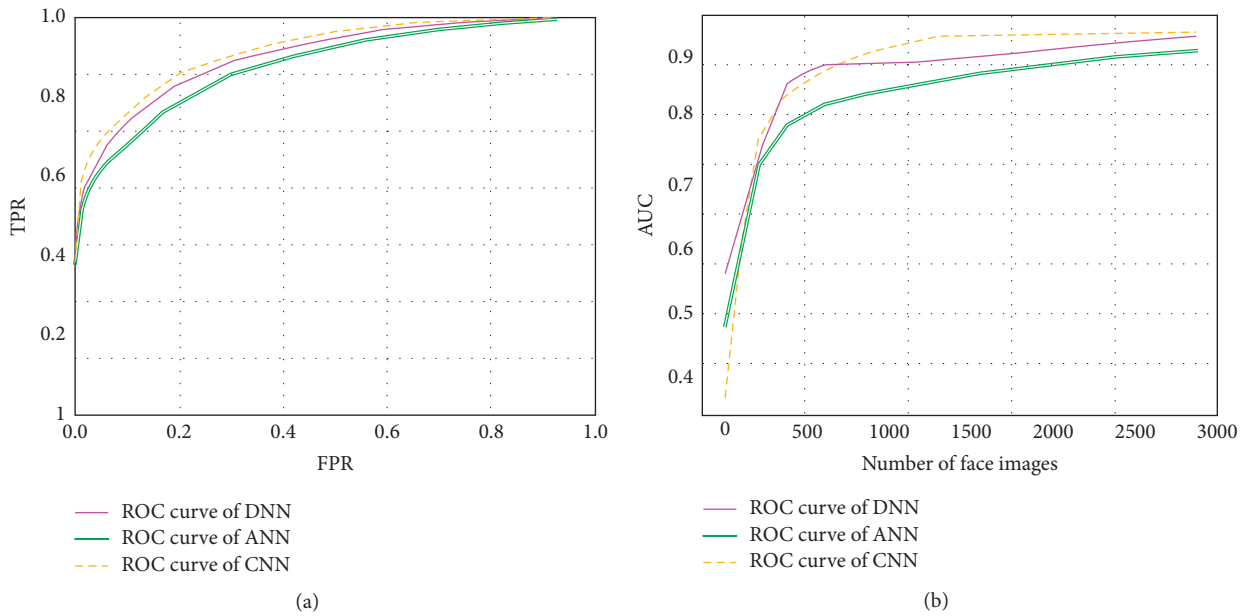


FIGURE 11: ROC curve and AUC curve of DNN, ANN, and CNN. (a) ROC curve of DNN, ANN, and CNN. (b) AUC curve of DNN, ANN, and CNN.

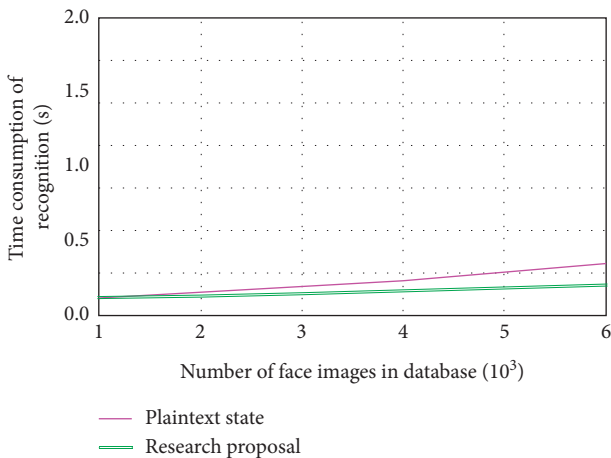


FIGURE 12: Time consumption in privacy preserving scheme.

which can save a lot of computing space. In the research process, the data space occupied by 10000 face images and 10000 face feature vectors are compared, and the results

show that the former occupies 85504.53 Kb. The latter only takes up 7031.21 kB of space, that is to say, the face feature vector data only take up about 10% of the space of the original face image. Therefore, edge computing is used to process the face image to improve the security of the intelligent system, and the face feature vector is used to replace the corresponding face image for subsequent operations, It can greatly reduce the storage pressure and communication load of intelligent system.

4. Conclusion

With the development of computer hardware technology, artificial intelligence technology ushered in the heyday of development, intelligent systems in various industries began to popularize, biometric identification has become the mainstream technology of intelligent system to achieve user identity authentication, but also an important part of measuring the security of intelligent system. To improve the security of intelligent system, a privacy protection scheme

based on edge computing, secure nearest neighbor, and secret sharing homomorphism is designed. The results show that with the increase of the number of iterations, the test loss value decreases and the test accuracy increases. When the number of iterations is 50000, the test loss value decreases to the minimum, the test accuracy reaches the maximum, and then gradually becomes stable; With the increase of the threshold, the face verification accuracy first increases and then decreases; When the threshold is 0.51, the correct rate of face verification reaches 92.46%, which is far higher than the recognition strength of human eyes; The ROC curves of ANN and DNN are all included in the range of CNN. The area under ROC curve of ANN and DNN was 0.8826 and 0.9278, respectively, which was less than that of CNN (0.9359). The convergence speed of the intelligent system based on CNN is faster than that based on ANN and DNN. The time consumption of the proposed algorithm is almost equal to that of the plaintext face recognition. Based on face feature vector data, only about 10% of the original face image space is needed. The above results show that the proposed privacy protection scheme based on edge computing can greatly improve the security of users using the intelligent system and effectively avoid user information leakage and data loss. In this research process, cosine similarity technology is used to measure the similarity of encrypted face feature vectors. The next step is to make full use of machine learning technology to accurately classify face feature vectors in ciphertext state. Although some achievements have been made in the research, the high-intensity demand of response time in application scenarios is not considered. In the future, encryption scheme should be further improved and response time should be shortened.

Data Availability

All the data in this study are from experimental data statistics.

Consent

Informed consent was obtained from all individual participants included in the study references.

Conflicts of Interest

The authors declare that there are no conflicts of interest.

Acknowledgments

This work is supported by Henan Science and Technology Plan Project (202102210355). Research on key technologies of CCN-based service deployment, discovery and scheduling optimization in MEC Environment.

References

- [1] P. Bagga, A. K. Das, and M. Wazid, "On the design of mutual authentication and key agreement protocol in internet of vehicles-enabled intelligent transportation system," *IEEE Transactions on Vehicular Technology*, vol. 99, p. 1, 2021.
- [2] H. Chen, C. C. Chang, and K. Chen, "Reversible data hiding schemes in encrypted images based on the paillier cryptosystem," *International Journal on Network Security*, vol. 22, no. 3, pp. 523–533, 2020.
- [3] C. Chen and X. Zhao, "Separate analysis of cell-edge and cell-centre user performance for irregular massive MIMO network with interference cancellation," *IET Communications*, vol. 13, no. 3, pp. 354–362, 2019.
- [4] Q. Feng, D. He, and S. Zeadally, "BPAS: blockchain-assisted privacy-preserving authentication system for vehicular Ad-Hoc networks," *IEEE Transactions on Industrial Informatics*, vol. 16, p. 4146, 2019.
- [5] T. M. Ghanim, M. I. Khalil, and H. M. Abbas, "Comparative study on deep convolution neural networks DCNN-based offline Arabic handwriting recognition," *IEEE Access*, vol. 99, p. 1, 2020.
- [6] S. J. Horng, J. Supardi, W. Zhou, C.-T. Lin, and B. Jiang, "Recognizing very small face images using convolution neural networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 99, pp. 1–13, 2020.
- [7] S. Huang, C. Yang, S. Yin, Z. Zhang, and Y. Chu, "Latency-aware task peer offloading on overloaded server in multi-access edge computing system interconnected by metro optical networks," *Journal of Lightwave Technology*, vol. 38, p. 1, 2020.
- [8] S. Jangirala, A. K. Das, M. Wazid, and A. V. Vasilakos, "Designing secure user authentication protocol for big data collection in IOT-based intelligent transportation system," *IEEE Internet of Things Journal*, vol. 8, p. 1, 2020.
- [9] A. Jolfaei, P. Ostovari, and M. Alazab, "Guest Editorial special issue on privacy and security in distributed edge computing and evolving IoT," *IEEE Internet of Things Journal*, vol. 7, no. 4, pp. 2496–2500, 2020.
- [10] C. F. Lai, W. C. Chien, L. T. Yang, and W. Qiang, "LSTM and edge computing for big data feature recognition of industrial electrical equipment," *IEEE Transactions on Industrial Informatics*, vol. 4, p. 1, 2019.
- [11] C. Lin, D. He, and X. Huang, "BCPPA: a blockchain-based conditional privacy-preserving authentication protocol for vehicular AD HOC networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 99, pp. 1–13, 2020.
- [12] C. Liu, X. Zhang, and Q. Hu, "Image super resolution convolution neural network acceleration algorithm," *Journal of National University of Defense Technology*, vol. 41, no. 2, pp. 91–97, 2019.
- [13] W. Liu, C. Qin, and K. Gao, "Research on medical data feature extraction and intelligent recognition technology based on convolutional neural network," *IEEE Access*, vol. 7, p. 1, 2019.
- [14] N. Padhy, R. K. Mishra, C. Satapathy, and K. Raju, "An automation API for authentication and security for file uploads in the cloud storage environment," *Intelligent Decision Technologies*, vol. 14, no. 3, pp. 393–407, 2020.
- [15] R. K. P. Varma, S. Ganta, B. H. Krishna, and S. Praveen, "A novel method for Indian vehicle registration number plate

- detection and recognition using image processing techniques,” *Procedia Computer Science*, vol. 167, pp. 2623–2633, 2020.
- [16] S. K. Sharma and B. Khuntia, “Integrated security for data transfer and access control using authentication and cryptography technique for Internet of things,” *International Journal of Knowledge-Based and Intelligent Engineering Systems*, vol. 24, no. 4, pp. 303–309, 2021.
- [17] C. Sonmez, C. Tunca, and A. Ozgovde, “Machine learning-based workload orchestrator for vehicular edge computing,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 4, pp. 2239–2251, 2021.
- [18] S. Sun, G. Zhang, and H. Mei, “Optimizing multi-uav deployment in 3d space to minimize task completion time in UAV-enabled mobile edge computing systems,” *IEEE Communications Letters*, vol. 25, p. 1, 2020.
- [19] Y. Tang, K. Guo, J. Ma, Y. Shen, and T. Chi, “A smart caching mechanism for mobile multimedia in information centric networking with edge computing,” *Future Generation Computer Systems*, vol. 95, pp. 590–600, 2019.
- [20] S. Wan, J. Lu, P. Fan, and K. B. Letaief, “Toward big data processing in IOT: path planning and resource management of UAV base stations in mobile-edge computing system,” *IEEE Internet of Things Journal*, vol. 7, no. 7, pp. 5995–6009, 2020.
- [21] X. Jiang, F. R. Yu, T. Song, and C. M. Leung, “Intelligent resource allocation for video analytics in blockchain-enabled internet of autonomous vehicles with edge computing,” *IEEE Internet of Things Journal*, vol. 99, p. 1, 2020.
- [22] X. Wang, D. Ma, and K. Hu, “Mapping based residual convolution neural network for non-embedding and blind image watermarking,” *Journal of Information Security and Applications*, vol. 59, no. 1, Article ID 102820, 2021.
- [23] L. Xiong, X. Zhong, N. N. Xiong, and R. W. Liu, “QR-3S: a high payload QR code secret sharing system for industrial internet of things in 6G networks,” *IEEE Transactions on Industrial Informatics*, vol. 17, p. 1, 2020.
- [24] X. Xu, Q. Liu, and Y. Luo, “A computation offloading method over big data for IoT-enabled cloud-edge computing,” *Future Generation Computer Systems*, vol. 95, pp. 522–533, 2019.
- [25] W. Zhang, X. Chen, Y. Liu, and Q. Xi, “A distributed storage and computation k-nearest neighbor algorithm based cloud-edge computing for cyber-physical-social systems,” *IEEE Access*, vol. 8, p. 1, 2020.
- [26] H. Zhu and L. Zhu, “Encrypted network behaviors identification based on dynamic time warping and k-nearest neighbor,” *Cluster Computing*, vol. 22, no. 1, pp. 1–10, 2019.

Research Article

SSGD: A Safe and Efficient Method of Gradient Descent

Jinhuan Duan,¹ Xianxian Li ,^{1,2} Shiqi Gao,² Zili Zhong,² and Jinyan Wang ^{1,2}

¹Guangxi Key Lab of Multi-Source Information Mining and Security, Guangxi Normal University, Guilin, China

²College of Computer Science and Engineering, Guangxi Normal University, Guilin, China

Correspondence should be addressed to Xianxian Li; lixx@gxnu.edu.cn and Jinyan Wang; wangjy612@gxnu.edu.cn

Received 30 May 2021; Revised 13 July 2021; Accepted 27 July 2021; Published 10 August 2021

Academic Editor: Lu Liu

Copyright © 2021 Jinhuan Duan et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

With the vigorous development of artificial intelligence technology, various engineering technology applications have been implemented one after another. The gradient descent method plays an important role in solving various optimization problems, due to its simple structure, good stability, and easy implementation. However, in multinode machine learning system, the gradients usually need to be shared, which will cause privacy leakage, because attackers can infer training data with the gradient information. In this paper, to prevent gradient leakage while keeping the accuracy of the model, we propose the super stochastic gradient descent approach to update parameters by concealing the modulus length of gradient vectors and converting it or them into a unit vector. Furthermore, we analyze the security of super stochastic gradient descent approach and demonstrate that our algorithm can defend against the attacks on the gradient. Experiment results show that our approach is obviously superior to prevalent gradient descent approaches in terms of accuracy, robustness, and adaptability to large-scale batches. Interestingly, our algorithm can also resist model poisoning attacks to a certain extent.

1. Introduction

Gradient descent (GD) is a technique to minimize an objective function, which is parameterized by the parameters of a model, by updating the parameters with the opposite direction of the gradient of the objective function about the parameters [1]. It has widely been applied in solving various optimization problems because of its simplicity and impressive generalization ability [2], but it is born with a heart of revealing privacy. Mathematically, the gradient is the parametric derivative of the loss function, which is explicitly calculated from the given training data and its true label. Therefore, the attacker may extract the sensitive information of the original training data from the captured gradients. Recently, researches have shown that the attacker, which captures the gradient of a training sample, can successfully infer its attributes [3], label [4], class representation [5, 6], or the data input itself [4, 7–9], with high accuracy. In the actual deep learning system, the gradient of multiple samples is widely used to improve efficiency and performance, which can also be viewed as the per-coordinate average of the single-sample gradients. Is multisample gradient safer for

the privacy of training data? Unfortunately, Pan et al. [9] gave the theoretical analysis to indicate that multisample gradient still leaks samples and labels under certain circumstances. Since the work of Zhu et al. [7] was proposed, there is a branch of research [4, 7–9] to explore a violent but universal method for successful data reconstruction attacks, and some meaningful empirical results are given on CIFAR-10 and ImageNet. These works are based on the same learning-based framework. First, a batch of unknown training samples are used as variables, and then the optimal training samples are searched by minimizing the distance between the ground-truth gradient and the gradient calculated by the variables. The main difference between them is the choice of minimizing distance function. L2 and cosine distances are used in [4, 7, 8], respectively. Although Zhao et al. [4] used the properties of neural networks to recover the label of a single sample before the learning-based attack, this technique is only suitable to single-point gradient. It is the same as [7] in the multisample case. Pan et al. [9] gave a theoretical explanation for information leakage of single sample in a fully connected neural network with ReLU activation function. Furthermore, they used the internal

information between neurons to show that in some cases there is sample and label leakage in multiple samples and extended the model to ResNet-18 [10], VGG-11 [11], DenseNet-121 [12], AlexNet [13], ShuffleNet v2-x0-5 [14], InceptionV3 [15], GoogLeNet [16], and MobileNet-V2 [17].

To solve the gradient safety problem, Bonawitz et al. [18] designed a secure aggregation protocol, which is a four-round interactive protocol. Xu et al. proposed VerifyNet [19] and VeriFL [20] by adding verifiability to [18] for ensuring the correctness of aggregation. Bell et al. [21, 22] introduced a secure aggregation protocol with multilogarithmic communication and computational complexity, which reduces one round of interaction compared with [18]. Fereidooni et al. [23] showed that only two rounds of communication can be safely aggregated. All of the above works use encryption algorithms to encrypt the entire data set or intermediate values during the training process. Different from them, Ma et al. [24] used secure verifiable computing delegation to privately label a public data set from locally trained model aggregation and then utilized public data sets to train local models. Phong et al. [25] used homomorphic encryption technology to encrypt the gradient before sending it. Abadi et al. [26] employed differential privacy to protect gradients. Yadav et al. [27] applied differential privacy to federated machine learning by directly adding noise to the gradient. In PrivateDL [28], it is allowed to effectively transfer relational knowledge from sensitive data to public data in a way of privacy protection and enables participants to jointly learn local models based on public data with noise protection labels. However, these methods also have their limitations. The main problem of the secure aggregation protocol is communication overhead and computational efficiency. For differential privacy technology, it needs to consider the tradeoff between privacy and utility. More noise will lead to poor performance, and less noise will not be enough to protect the gradient. PrivateDL [28] requires a public data set and reduces the performance of the algorithm.

Therefore, this paper proposes a new gradient descent method, super stochastic gradient descent (SSGD), for achieving neuron-level security while maintaining the accuracy of model. Moreover, SSGD has stronger robustness. Phong et al. [25] analyzed the leakage of single-sample single-neuron input data in the single-layer perceptron by using the sigmoid activation function. Pan et al. [9] used the ReLU activation function to analyze the sample data leakage from the multilayer fully connected neural network gradient and indicated that multiple samples also reveal privacy. There are two neurons in the last layer which are only activated by the same single sample. Essentially, the leakage is caused by attacking the single-sample gradient. SSGD converts the neuron gradient into a unit vector, which makes that the gradient aggregation of neurons has super-randomness. Superrandomness may significantly worsen the performance of the algorithm and make it difficult to converge. We select multiple-sample gradient composition updates to increase stability. At the same time, the super-randomness also brings strong robustness because the attacker cannot know the true gradient. SSGD invalidates

these attacks on the gradient model, including the attack by searching for the optimal training sample [4, 7, 8] based on minimizing the distance between the ground-truth gradient and the gradient calculated by the variable, and the attack by solving the equation system [9] to obtain the training data. Our contributions are summarized as follows.

- (1) We propose a gradient descent algorithm, called super stochastic gradient descent. The main idea is to update the parameters by using the unit gradient vector. In neural networks, neuron parameters are updated by using the unit gradient vector of neurons.
- (2) We analyze theoretically that SSGD can realize neuron-level security and defend against attacks on the gradient.
- (3) Experimental results show our approach has better accuracy and robustness than prevalent gradient descent approaches. And it can resist model poisoning attacks to a certain extent.

The rest of this paper is organized as follows. In Section 2, we review the basic gradient descent methods and the data leakage by gradients. In Section 3, we describe the super stochastic gradient descent and analyze the safety of our approach. The experimental results are shown in Section 4. Finally, we conclude this paper and give the further work.

2. Preliminaries

In this section, we review some basic gradient descent algorithms [1], including batch gradient descent (BGD), stochastic gradient descent (SGD), and mini-batch gradient descent (MBGD). The difference among them is that how much data is used to calculate the gradient of the objective function. Then, we describe the information leakage caused by gradients [19].

2.1. Basic Gradient Descent Algorithms. The BGD is an ordinary form of gradient descent, which takes the entire training samples into account to calculate the gradient of the cost function $\ell(\theta)$ about the parameters θ and then update the parameters by

$$\theta = \theta - \eta \cdot \nabla_{\theta} \ell(\theta), \quad (1)$$

where η is the learning rate and $\nabla_{\theta} \ell(\theta)$ represents the gradient of function $\ell(\theta)$ with respect to the parameters θ . The BGD uses the entire training set in each iteration. Therefore, the update is proceeded in the right direction, and finally BGD is guaranteed to converge to the extreme point. On the contrary, the SGD considers a training sample x_i and label y_i randomly selected from the training set in each iteration to perform the update of parameters by

$$\theta = \theta - \eta \cdot \nabla_{\theta} \ell(\theta; x_i; y_i). \quad (2)$$

The BGD and SGD are two extremes: one uses all training samples and the other uses one sample for gradient descent. Naturally, their advantages and disadvantages are very prominent. For the training speed, the SGD is very fast,

and the BGD cannot be satisfactory when the size of training sample set is large. For accuracy, the SGD determines the direction of the gradient with only one sample, resulting in a solution which may not be optimal. For the convergence rate, because the SGD considers one sample in each iteration and the gradient direction changes greatly, it cannot quickly converge to the local optimal solution.

The MBGD is a compromise between BGD and SGD, which performs an update with a randomly sampled mini-batch of N training samples by

$$\theta = \theta - \eta \cdot \nabla_{\theta} \ell(\theta; \mathbf{x}_{(i:i+N)}; \mathbf{y}_{(i:i+N)}), \quad (3)$$

where N is the number of batches. MBGD decreases the variance of the updates for parameter, so it has more stable convergence. Moreover, the computing of gradient about a mini-batch is very efficient by using highly optimized matrix optimizations that existed in advanced deep learning libraries.

2.2. Analysis of Gradient Information Leakage. Phong et al. [25] illustrated that how gradients leak the data information based on a single neuron shown in Figure 1. Assume that $\hat{\mathbf{x}} \in R^d$ represents data input with a label value $y \in R$. $w \in R^d$ is the weight parameter and $b \in R$ is the bias, represented uniformly by $\theta = (w, b) \in R^{(d+1)}$. $g \in R^{(d+1)}$ is the gradient vector of the parameter θ , f is an activation function, and the loss function is $\ell(f(\hat{\mathbf{x}}, w, b), y) = (h_{w,b}(\hat{\mathbf{x}}) - y)^2$, where $h_{w,b}(\hat{\mathbf{x}}) = f(\sum_{i=1}^d w_i \hat{x}_i + b)$. Let $g = (\sigma_1, \dots, \sigma_k, \dots, \sigma_d, \sigma)$ and $k \in \{1, \dots, d\}$. We have

$$\begin{aligned} \sigma_k &= \frac{\partial \ell(f(\hat{\mathbf{x}}, w, b), y)}{\partial w_k} = 2(h_{w,b}(\hat{\mathbf{x}}) - y) f' \left(\sum_{i=1}^d w_i \hat{x}_i + b \right) \cdot \hat{x}_k, \\ \sigma &= \frac{\partial \ell(f(\hat{\mathbf{x}}, w, b), y)}{\partial b} = 2(h_{w,b}(\hat{\mathbf{x}}) - y) f' \left(\sum_{i=1}^d w_i \hat{x}_i + b \right). \end{aligned} \quad (4)$$

Therefore, we obtain $\sigma_k = \sigma \cdot \hat{x}_k$. By solving the system of equations, we can easily get $\hat{\mathbf{x}}$ and y . Also, we know that g is determined by $(\hat{\mathbf{x}}, y)$. Therefore, g and $(\hat{\mathbf{x}}, y)$ are bijective. In distributed training, w and b usually are the parameters that need to be updated and known. Then, it can infer $(\hat{\mathbf{x}}, y)$ from g .

Based on [9], the single-sample analysis of multilayer neural networks by using ReLU activation function, there is also data leakage problem. Although there is no such simple and intuitive leakage of data in a multilayer neural network, we can still know $\hat{\mathbf{x}}$ and y by analyzing the internal relationship of the neural network and find that $(\hat{\mathbf{x}}, y)$ and g are still bijective.

3. Super Stochastic Gradient Descent

In this section, we propose our super stochastic gradient descent approach for preventing gradient leakage while keeping the accuracy and then analyze in detail the safety of our approach.

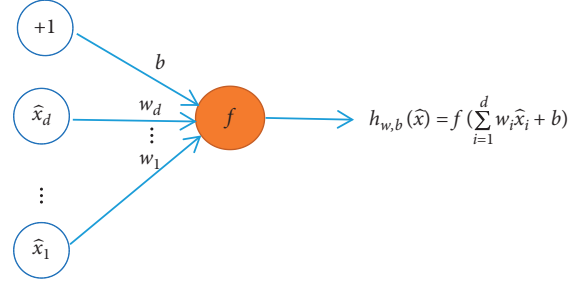


FIGURE 1: Single neuron structure.

3.1. Approach. It was confirmed that the gradient leaks privacy [7, 25]. For solving the security problem caused by the exchange gradient in stochastic gradient descent or mini-batch gradient descent, we propose the super stochastic gradient descent approach, which can protect the gradient information without losing accuracy by hiding part of the gradient information. The gradient is the first-order partial derivative of the objective function, so it is a vector with both magnitude and direction. We seek the gradient of the objective function to find the fastest descent direction. But it is a little related to the modulus length of the gradient vector. Therefore, we hide the modulus length of the gradient vector and convert the gradient vector into a unit vector.

The superrandomness, caused by the aggregation of multiple unit gradient vectors, may lead to poor results. To guarantee that this kind of randomness is friendly, we utilize the following approaches to reduce the uncertainty caused by superrandomness.

For single-sample training sample x_i and label y_i , we use unit gradient vector to update parameter θ :

$$\theta = \theta - \eta \cdot \frac{\nabla_{\theta} \ell(\theta; x_i; y_i)}{\|\nabla_{\theta} \ell(\theta; x_i; y_i)\|}. \quad (5)$$

For multiple samples, the parameter is updated to

$$\theta = \theta - \frac{\eta}{m} \cdot \sum_{j=1}^m \frac{\nabla_{\theta} \ell(\theta; x_{(i:i+n)}; y_{(i:i+n)})_j}{\|\nabla_{\theta} \ell(\theta; x_{(i:i+n)}; y_{(i:i+n)})_j\|}, \quad (6)$$

where $x_{(i+n)}$ represents n samples and $y_{(i+n)}$ denotes their labels. The gradient $\nabla_{\theta} \ell(\theta; x_{(i:i+n)}; y_{(i:i+n)})$ of n samples is considered as a basic gradient, and m is the number of basic gradients. Aggregating the unit gradient vectors of m basic gradients on average is to further enhance the stability of the algorithm. The algorithm has higher performance with strong randomness. It is secure to share this unit basic gradient in a distributed environment.

Neuron is the smallest information carrier in the neural network structure. In the neural network, we choose to convert each neuron parameter gradient vector into a unit vector. Therefore, the single-layer neural network parameter is updated to

$$\theta_r = \theta_r - \frac{\eta}{m} \cdot \sum_{j=1}^m \frac{\nabla_{\theta} \ell(\theta; x_{(i:i+n)}; y_{(i:i+n)})_{rj}}{\|\nabla_{\theta} \ell(\theta; x_{(i:i+n)}; y_{(i:i+n)})_{rj}\|}, \quad (7)$$

where θ_r represents the r th column or r th row of the parameter matrix in the fully connected layer or convolutional layer (the convolution kernel is regarded as a neuron). In the fully connected layer, $\nabla_{\theta} \ell(\theta; x_{(i:i+n)}; y_{(i:i+n)})_r$ is expressed as the r th column of the gradient matrix. And in the convolutional layer, it represents the r th row of the gradient matrix of the convolution kernel. Therefore, each row or column of the gradient matrix is a unit vector. Then, we obtain an average gradient matrix by using m such gradient matrices to update the parameters.

3.2. The Safety of SSGD. By analyzing the multilayer neural network with ReLU activation function on a training sample, the following relationship is obtained in [9]:

$$\bar{G}^i = \sum_c \bar{g}_c (D^i W^{i-1} \dots W^0 X) \left([W^H]_c^T D^H \dots W^{(i+1)} D^{(i+1)} \right), \quad (8)$$

in which $X = \{x_1, x_2, \dots, x_n\}$ is the input data, where $x_i \in R^d$ and $X \in R^{d \times n}$. \bar{g}_c represents the c th dimension of the loss vector \bar{g} , T is the number of layers of neural network, D^i is the activation pattern of the i th layer of neural network, and \bar{G}^i and W^i denote the gradients and parameters of the i th layer of neural network, respectively. In fact, the attack gradient models are all solutions to the above equations. In the distributed training model that needs to share the gradient, the participants know \bar{G}^i , W^i , and D^i . For data reconstruction attacks, it can infer \bar{g}_c and solve X by equation (8).

The left side of equation (8) is the i th layer gradient matrix:

$$\bar{G}^i = \begin{bmatrix} \sigma_{1,1}^i & \sigma_{1,2}^i & \dots & \sigma_{1,w^i}^i \\ \sigma_{2,1}^i & \sigma_{2,2}^i & \dots & \sigma_{2,w^i}^i \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{w^{i-1},1}^i & \sigma_{w^{i-1},2}^i & \dots & \sigma_{w^{i-1},w^i}^i \end{bmatrix}_{w^{i-1} \times w^i}, \quad (9)$$

where w^i is the number of neurons in the i th layer. The gradient matrix of our SSGD is

$$\hat{G}^i = \begin{bmatrix} \sigma_{1,1}^i & \sigma_{1,2}^i & \dots & \sigma_{1,w^i}^i \\ \sigma_{2,1}^i & \sigma_{2,2}^i & \dots & \sigma_{2,w^i}^i \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{w^{i-1},1}^i & \sigma_{w^{i-1},2}^i & \dots & \sigma_{w^{i-1},w^i}^i \end{bmatrix}_{w^{i-1} \times w^i} \begin{bmatrix} \frac{1}{\mu_1^i} & 0 & \dots & 0 \\ 0 & \frac{1}{\mu_2^i} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \frac{1}{\mu_{w^i}^i} \end{bmatrix}_{w^i \times w^i}. \quad (10)$$

Each column of \hat{G}^i is a unit vector, and μ_1^i is the modulus length of the 1st column vector of the i th layer gradient matrix, i.e., the modulus length of the 1st neuron gradient of the i th layer neural network. Essentially, the parameter matrix of a layer of neural network is multiplied by a

diagonal matrix U^i on the right, and the value of the diagonal matrix is the reciprocal of the modulus length of the gradient vector of each neuron. By using our SSGD, equation (8) is represented as

$$\hat{G}^i = \sum_c \bar{g}_c (D^i W^{i-1} \dots W^0 X) \left([W^H]_c^T D^H \dots W^{i+1} D^{i+1} \right) U^i, \quad (11)$$

where U^i is unknown and is not uniquely determined when the loss functions are nonconvex and nonconcave functions. According to [29], we know that the loss function of multilayer neural networks are nonconvex and nonconcave functions. Due to the dynamicity of U^i , even if \bar{g}_c , \hat{G}^i , W^i , and D^i are known, X is not obtained.

Our method hides the correlation between the gradient and the sample, eliminates the information between neurons, and achieves neuron-level security. SSGD is a multi-sample training; there is no information leakage problem like in [19], which is a single-sample leakage of privacy. SSGD can defend against attacks on the gradient.

Since training a model requires rounds of iterations, is it safe to use multiple rounds of iterations? We previously analyzed that the gradient g and the training data (\hat{x}, y) are bijective in terms of parameter θ , i.e., $g = \nabla_{\theta} f(\theta | (\hat{x}, y))$, where f is a functional relationship. We use θ^i and θ^{i+1} to denote the training parameters of the i th and $i+1$ st rounds, respectively. Then, we have $\theta^{(i+1)} = \theta^i - \eta \cdot g^i$. The i th gradient $g^i = \nabla_{\theta} f(\theta^i | (\hat{x}, y))$. Therefore, we have $\theta^{(i+1)} = \theta^i - \eta \cdot \nabla_{\theta} f(\theta^i | (\hat{x}, y))$. Furthermore, we obtain $g^{(i+1)} = \nabla_{\theta} f(\theta^i - \eta \cdot \nabla_{\theta} f(\theta^i | (\hat{x}, y)) | (\hat{x}, y))$. By comparing $g^{(i+1)}$ with g^i , we can see that there is not additional information in $g^{(i+1)}$. The information of the model is only related to the training samples, initial parameters, and learning rate. Therefore, the iteration operation does not cause the information leakage.

4. Experiments

Data. We use MNIST (<https://yann.lecun.com/exdb/mnist>) and Fashion-MNIST (<https://fashion-mnist.s3-website-eu-central-1.amazonaws.com>) datasets to assess the performance of our algorithm. The MNIST contains 60000 training images and 10000 test images, where every image is a 28×28 grayscale image, and each pixel is an octet. The Fashion-MNIST [30] is composed of 28×28 grayscale images of 70,000 fashion products from 10 categories, with 7,000 images per category. The training set and test set contain 60,000 images and 10,000 images, respectively.

Model. The lenet-5 [31] contains two convolutional layers, two pooling layers, and three fully connected layers. The activation function is ReLU. The input dimensions are 784, and output dimensions are 10.

Evaluation Index (The Test Accuracy). We use 60000 training images to train model. The test accuracy is the average value of ten experimental results, and every experiment obtains

the average test accuracy of randomly selecting 1000 samples from the test set. The number of iterations is 10,000. The highest test accuracy of these compared algorithms in the same experimental environment is shown in bold.

4.1. Accuracy and Efficiency. We compare SSGD with SGD, SGDm [32], and Adam [33], which are widely used gradient descent algorithms. The batch size ($N = m \times n$) is set to 16, 32, 64, 128, 256, 512, 1024, 2048, 4096, and 8192, where n is set to 1, 4, 8, 16, 32, 64, and 128 and m is set to 4, 8, 16, 32, and 64. When $m = 1$, it is the MBGD. There is not set same learning rate as a good experimental result, because SGD and SGDm have poor adaptability in large batches.

For MNIST data set, the momentum of SGDm is set to 0.999. For the experimental parameters of Adam, the learning rate is set to 5×10^{-4} , and β_1 and β_2 are set to 0.9 and 0.999, respectively. For SSGD, the learning rate in this experiment is set to 10^{-1} . For Fashion-MNIST data set, the momentum of SGDm is set to 0.99. For the experimental parameters of Adam, the learning rate is set to 10^{-3} , and β_1 and β_2 are set to 0.9 and 0.999, respectively. For SSGD, the learning rate in this experiment is set to $10^{-2}/1.5^{0.002j}$, where j is the number of iterations.

The comparative experimental results of SGD, SGDm, Adam, and SSGD are shown in Tables 1 and 2, where the numbers in bracket in the second and third columns denote the learning rates of SGD and SGDm, respectively, and the number in bracket in the fifth column is the value of m . From Tables 1 and 2, we can see that the performance of our algorithm is better than that of SGD, SGDm, and Adam for large batches of data. In this case, SGD and SGDm need to reduce the learning rate to adapt to it. And Adam also has obvious overfitting in large batches of data. SSGD has always maintained high precision. On the whole, our algorithm on test accuracy is better and more stable than SGD, SGDm, and Adam.

Tables 3 and 4 show the running results of our SSGD approach in different numbers of training batches. We can see that the larger the number of training batches ($N = m \times n$) is, the better the test accuracy is. When the number of training batches is too small, the effect of n on performance is greater than that of m . The distribution of m values in Tables 1 and 2 also shows this point.

The convergence rate graphs on MNIST and Fashion-MNIST are shown in Figures 2(a) and 2(b), respectively. The value in longitudinal axis is the average accuracy of every 10 iterations. The SSGDm is SSGD with momentum. We choose the intermediate value 256 as the batch number in the convergence experiment, where $n = 16$ and $m = 16$ for SSGD and SSGDm. In Figure 2(a), the learning rates of SGD and SGDm are 10^{-4} and 5×10^{-4} , respectively. The momentum of SGDm is set to 0.999. The learning rate of SSGDm is $10/1.0002^j$, where j is the number of iterations, and its momentum is 0.99. The other parameters are consistent with the above experiment on MNIST. In Figure 2(b), we choose the larger batch number 1024 as the batch number in the convergence experiment, where $n = 64$ and $m = 16$ for SSGD and SSGDm. The learning rates of SGD and SGDm are 10^{-5} and 10^{-3} , respectively. The momentum of

TABLE 1: The test accuracy of compared algorithms on MNIST.

$N = m \times n$	SGD (η)	SGDm (η)	Adam	SSGD (m)
16	0.9781 (5×10^{-4})	0.9778 (5×10^{-4})	0.9767	0.9832 (4)
32	0.9702 (5×10^{-4})	0.9768 (5×10^{-4})	0.9850	0.9876 (8)
64	0.9794 (5×10^{-4})	0.9792 (5×10^{-4})	0.9842	0.9900 (8)
128	0.9676 (5×10^{-4})	0.9780 (5×10^{-4})	0.9896	0.9901 (16)
256	0.9755 (10^{-4})	0.9804 (5×10^{-4})	0.9855	0.9877 (16)
512	0.9665 (10^{-4})	0.9789 (5×10^{-4})	0.9814	0.9861 (16)
1024	0.9738 (10^{-5})	0.9749 (10^{-4})	0.9778	0.9869 (16)
2048	0.9763 (10^{-5})	0.9717 (10^{-4})	0.9894	0.9886 (64)
4096	0.9703 (10^{-5})	0.9806 (10^{-4})	0.9788	0.9878 (64)
8192	0.9785 (2×10^{-6})	0.8994 (10^{-4})	0.9753	0.9855 (64)

TABLE 2: The test accuracy of compared algorithms on fashion-MNIST.

$N = m \times n$	SGD (η)	SGDm (η)	Adam	SSGD (m)
16	0.8253 (10^{-4})	0.7795 (10^{-3})	0.7175	0.8035 (4)
32	0.8241 (10^{-4})	0.8031 (10^{-3})	0.7513	0.8046 (4)
64	0.8468 (10^{-4})	0.8163 (10^{-3})	0.7674	0.8344 (4)
128	0.8629 (10^{-4})	0.8331 (10^{-3})	0.7835	0.8437 (4)
256	0.8527 (10^{-4})	0.8511 (10^{-3})	0.8252	0.8602 (4)
512	0.8682 (10^{-4})	0.8569 (10^{-3})	0.8566	0.8590 (4)
1024	0.8569 (10^{-5})	0.8612 (10^{-3})	0.8547	0.8668 (16)
2048	0.8457 (10^{-5})	0.8351 (10^{-4})	0.8511	0.8652 (32)
4096	0.8629 (10^{-6})	0.8518 (10^{-4})	0.8321	0.8704 (32)
8192	0.8325 (10^{-6})	0.8278 (10^{-4})	0.8144	0.8648 (64)

TABLE 3: Test accuracy of SSGD on MNIST.

	$n = 1$	$n = 4$	$n = 8$	$n = 16$	$n = 32$	$n = 64$	$n = 128$
$m = 4$	0.9717	0.9832	0.9842	0.9846	0.9895	0.9886	0.9888
$m = 8$	0.9815	0.9876	0.9900	0.9884	0.9855	0.9861	0.9849
$m = 16$	0.9801	0.9854	0.9901	0.9877	0.9861	0.9869	0.9832
$m = 32$	0.9851	0.9804	0.9901	0.9833	0.9819	0.9867	0.9828
$m = 64$	0.9830	0.9849	0.9831	0.9833	0.9886	0.9878	0.9855

TABLE 4: Test accuracy of SSGD on Fashion-MNIST.

	$n = 1$	$n = 4$	$n = 8$	$n = 16$	$n = 32$	$n = 64$	$n = 128$
$m = 4$	0.7739	0.8035	0.8046	0.8344	0.8437	0.8602	0.8590
$m = 8$	0.8152	0.8136	0.8150	0.8271	0.8401	0.8564	0.8655
$m = 16$	0.8137	0.8176	0.8246	0.8446	0.8446	0.8668	0.8663
$m = 32$	0.8189	0.8125	0.8198	0.8353	0.8574	0.8652	0.8704
$m = 64$	0.8215	0.8196	0.8238	0.8289	0.8577	0.8655	0.8648

SGDm is set to 0.99. The other parameters are consistent with the above experiment on Fashion-MNIST. The learning rate of SSGDm is $1/1.5^{0.002j}$, where j is the number of iterations, and its momentum is 0.9. From Figure 2, we can see

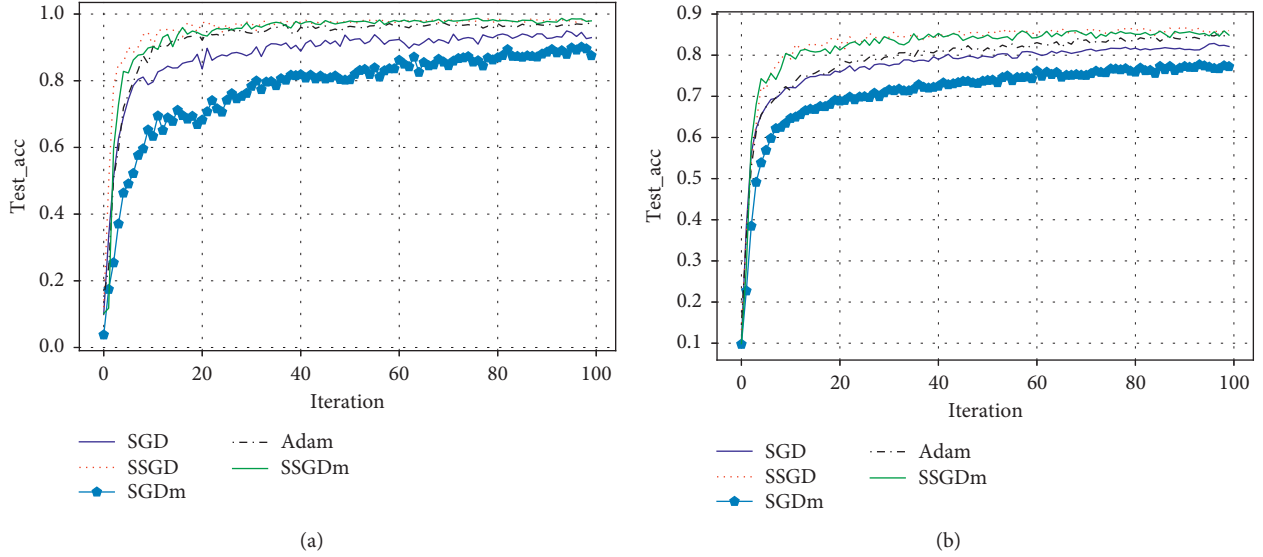


FIGURE 2: The convergence speed of test accuracy (a) on MNIST and (b) on Fashion-MNIST.

that the convergence speed of our algorithm is faster and more stable than SGD, SGDm, and Adam.

4.2. Robustness. Robustness is the robustness of the system, which refers to the characteristic that the system maintains a certain performance under certain parameter perturbations. To check the robustness of our algorithm, we add random noise to the gradient. At the same time, we noticed that differential privacy is a way to protect gradient information by adding random noise that meets a certain distribution. To compare the performances of our algorithm and the model with differential privacy, we choose the model in the robustness experiment to add noise that satisfies differential privacy. In this section, we compare the performances of the traditional gradient descent algorithm and SSGD with noises. In [20], the large gradient does not participate in the update, which will seriously affect the gradient descent performance. However, the large gradient participating in the update will cause the noise scale to be too large, which makes the algorithm effect extremely poor or even unable to converge. Different from cutting gradient value in [20], we strictly define sensitivity as the maximum value minus the minimum value in the gradient matrix. We add Laplacian noises of the same scale on comparing algorithms and set privacy budget $\epsilon=4$ and $\epsilon=2$ on MNIST and Fashion-MNIST, respectively.

We use SGDm and Adam as the compared algorithms. Also, we have tested SGD algorithm. When noise or the number of batches is large, the gradient explosion will occur and the SGD cannot converge on MNIST. SGDm and Adam algorithms have better robustness. Because both SGDm and Adam have momentum, SSGDm is chosen as our comparison algorithm. We adjust hyper parameters to get more performance for SGDm and Adam with noises. To make SGDm, Adam, and SSGDm experiments in the same environment, the batch number is $N = n \times m$, where n is set to 4, 8, 16, 32, and 64, and m is set

to 4, 8, 16, 32, and 64. For each iteration, after the n vectors are added, the Laplace noises of $\epsilon=4$ or $\epsilon=2$ that strictly meet the differential privacy are added. The sensitivity is set to the maximum value minus the minimum value of the gradient matrix of the same batch. Then, we use SGDm, Adam, and SSGDm algorithms to update their parameters, respectively. For SGDm, the momentum is 0.99. The learning rate is 10^{-2} and 10^{-3} on MNIST and Fashion-MNIST, respectively. For Adam, the learning rate is 10^{-3} on MNIST and Fashion-MNIST, $\beta_1=0.9$ and $\beta_2=0.999$. For SSGDm, we use the average of multiple-unit gradient vectors to update the gradient. Therefore, the module length of the update gradient vector decreases very slowly, and dynamic learning rates need to be set. The learning rate of SSGDm is set to $10/1.0002^j$ and $1/1.5^{0.002j}$ on MNIST and Fashion-MNIST, respectively. The momentum = 0.9.

From Tables 5 and 6, all three algorithms comply with the law of acquaintance; that is, the larger the batch size is, the better the accuracy is. We can see that SSGDm is more robust than the SGDm and Adam algorithms when the noises of the same scale are added in gradients on test accuracy. On MNIST, compared with SGDm and Adam, the average test accuracy of SSGDm is increased by 4.12% and 1.60%, respectively. On Fashion-MNIST, compared with SGDm and Adam, the average test accuracy of SSGDm is increased by 5.24% and 1.64%, respectively.

Where is the limit of the robustness of our algorithm? On MNIST, we try to increase the scale of noises and make ϵ be 0.2, 0.5, 1, 2, and 4. The experimental environment is the same as the robustness experiment above, and the parameter settings are also the same. The batch number is set to $n=16$ and $m=16$. On Fashion-MNIST, we make ϵ be 0.5, 1, 2, and 4. The batch number is set to $n=64$ and $m=16$. From Tables 7 and 8, it is clear that our SSGDm has obvious advantages in robustness. The greater the scale of noises is, the more obvious the advantage of our algorithm is.

TABLE 5: Test accuracy with $\varepsilon = 4$ on MNIST.

SGDm\Adam\SSGDm	$n = 4$	$n = 8$	$n = 16$	$n = 32$	$n = 64$
$m = 4$	0.8714\0.9321\0.9730	0.9299\0.9515\0.9816	0.9508\0.9592\0.9822	0.9559\0.9567\0.9774	0.9518\0.9715\0.9851
$m = 8$	0.9242\0.9570\0.9785	0.9337\0.9657\0.9829	0.9582\0.9737\0.9790	0.9569\0.9569\0.9832	0.9625\0.9797\0.9802
$m = 16$	0.9471\0.9607\0.9684	0.9390\0.9606\0.9839	0.9662\0.9723\0.9833	0.9674\0.9726\0.9844	0.9699\0.9796\0.9860
$m = 32$	0.9203\0.9580\0.9780	0.9333\0.9693\0.9805	0.9594\0.9658\0.9859	0.9647\0.9763\0.9838	0.9758\0.9806\0.9837
$m = 64$	0.9163\0.9545\0.9772	0.9514\0.9771\0.9861	0.9560\0.9715\0.987	0.9710\0.9702\0.9853	0.9678\0.9833\0.9885

TABLE 6: Test accuracy with $\varepsilon = 2$ on Fashion-MNIST.

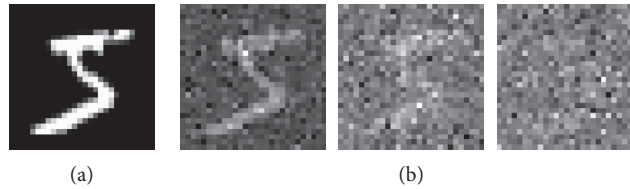
SGDm\Adam\SSGDm	$n = 4$	$n = 8$	$n = 16$	$n = 32$	$n = 64$
$m = 4$	0.7089\0.7177\0.7732	0.7623\0.7729\0.7875	0.7894\0.7813\0.7947	0.7941\0.8180\0.8233	0.8108\0.8192\0.8343
$m = 8$	0.7123\0.7402\0.7885	0.7658\0.7548\0.8040	0.7806\0.8022\0.8175	0.8044\0.8242\0.8412	0.8159\0.8373\0.8459
$m = 16$	0.7127\0.7723\0.7903	0.7763\0.8052\0.8139	0.7724\0.8173\0.8389	0.7993\0.8291\0.8474	0.8199\0.8386\0.8455
$m = 32$	0.7159\0.8013\0.8177	0.7669\0.8066\0.8333	0.7735\0.8339\0.8467	0.8103\0.8540\0.8611	0.8299\0.8553\0.8597
$m = 64$	0.7158\0.8140\0.8235	0.7432\0.8309\0.8412	0.7853\0.8497\0.8524	0.8064\0.8586\0.8647	0.8283\0.8663\0.8655

TABLE 7: The test accuracy by varying ε on MNIST.

	$\varepsilon = 0.2$	$\varepsilon = 0.5$	$\varepsilon = 1$	$\varepsilon = 2$	$\varepsilon = 4$
SGDm	0.0970	0.3745	0.8166	0.9344	0.9662
Adam	0.8074	0.8813	0.9140	0.9416	0.9723
SSGDm	0.8481	0.9395	0.9671	0.9719	0.9833

TABLE 8: The test accuracy by varying ε on fashion-MNIST.

	$\varepsilon = 0.5$	$\varepsilon = 1$	$\varepsilon = 2$	$\varepsilon = 4$
SGDm	0.0995	0.5921	0.8199	0.8363
Adam	0.6474	0.7934	0.8386	0.8524
SSGDm	0.7587	0.8106	0.8455	0.8618

FIGURE 3: Training sample image of MNIST. (a) The original image. (b) From left to right are the poisoned images with $\varepsilon = 5, 2,$ and $1,$ respectively.TABLE 9: Test accuracy by varying ε on MNIST.

	$\varepsilon = 1$	$\varepsilon = 2$	$\varepsilon = 5$
SGD	0.1095	0.9171	0.9679
Adam	0.4859	0.8160	0.8890
SSGD	0.9446	0.9621	0.9827

TABLE 10: Test accuracy by varying ε on Fashion-MNIST.

	$\varepsilon = 1$	$\varepsilon = 2$	$\varepsilon = 5$
SGD	0.1012	0.5813	0.7969
Adam	0.2888	0.3452	0.6296
SSGD	0.4638	0.7651	0.8290

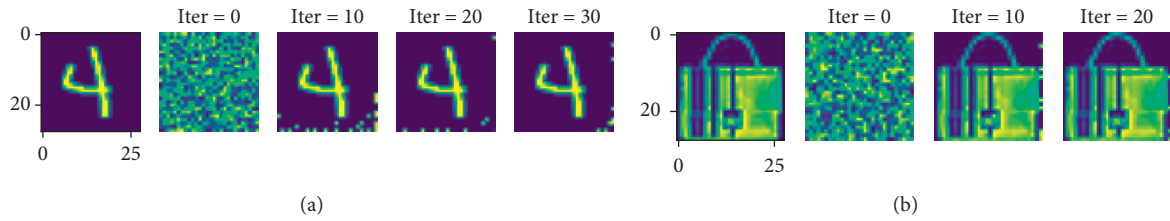


FIGURE 4: DLG attacks SGD (a) on MINST dataset and (b) on Fashion-MNIST dataset.

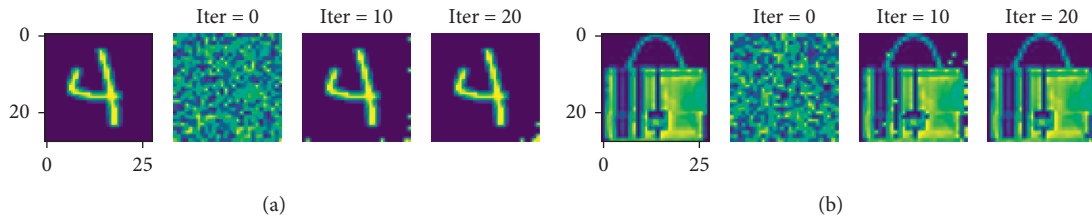


FIGURE 5: iDLG attacks SGD (a) on MINST dataset and (b) on Fashion-MNIST dataset.

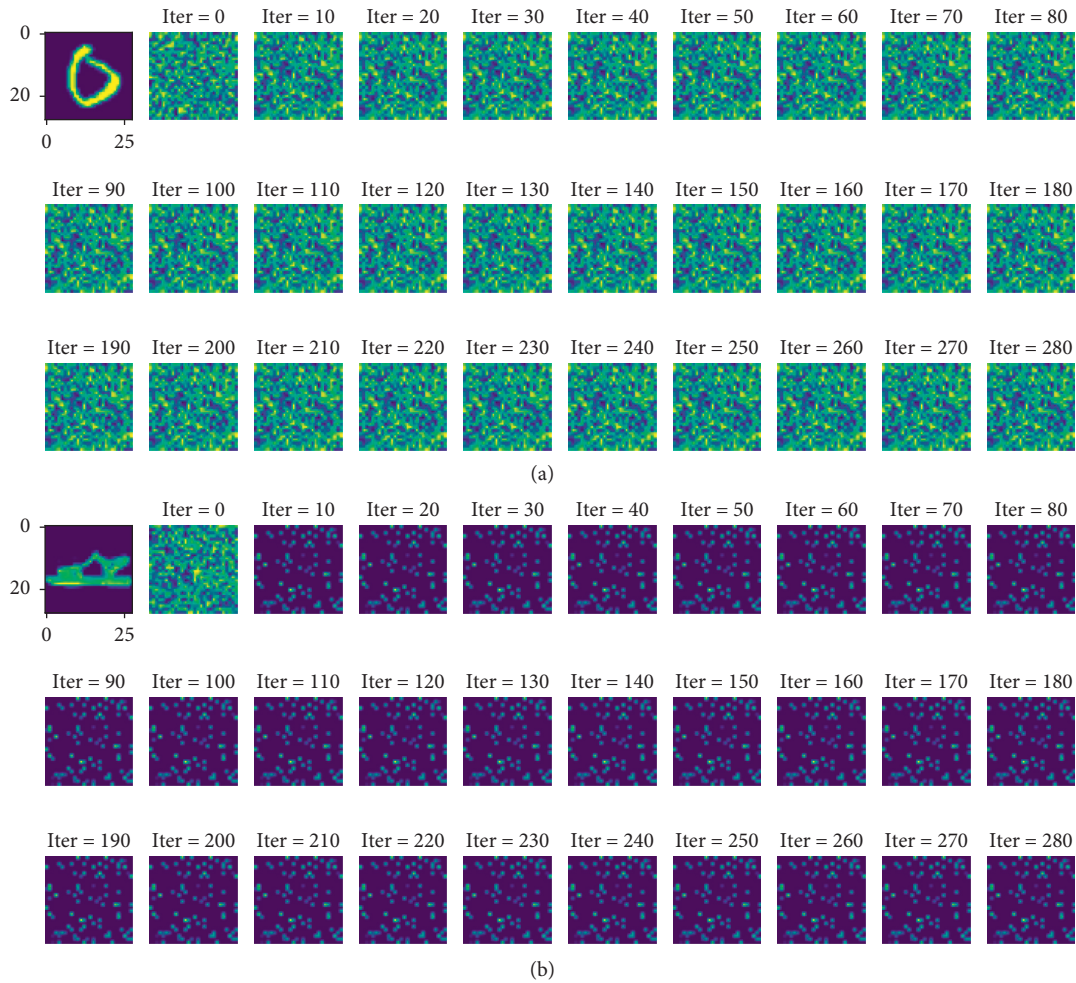


FIGURE 6: DLG attacks SSGD (a) on MINST dataset and (b) on Fashion-MNIST dataset.

4.3. Poisoning Attack. The goal of poisoning attack is to destroy the integrity and availability of data. The robustness experiment results show that our algorithm can resist the

poisoning attack added to the gradient to a certain extent. According to the previous analysis, the gradient is a kind of mapping of the training data. Then, our algorithm should be

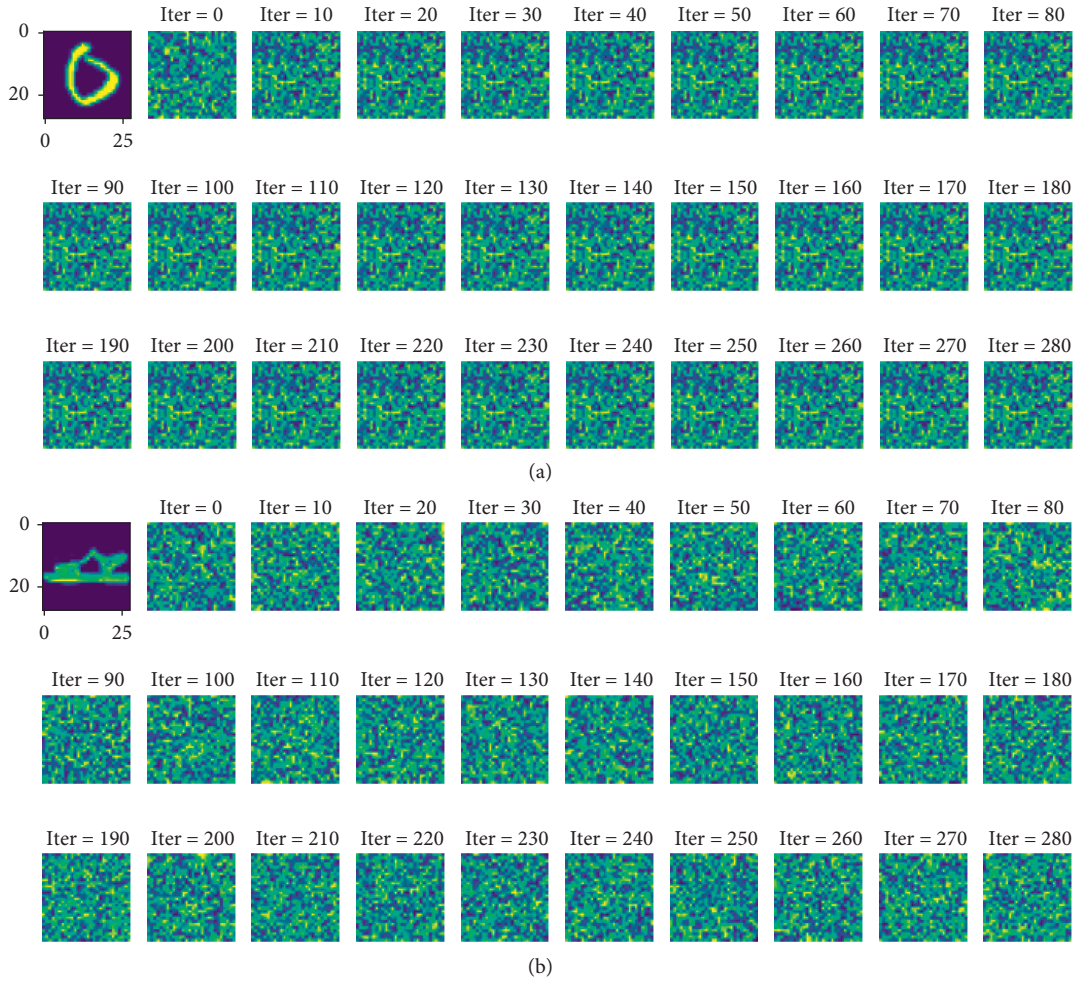


FIGURE 7: iDLG attacks SSGD (a) on MNIST dataset and (b) on Fashion-MNIST dataset.

effective against data poisoning attacks. This part of the experiment is to verify the performance of our algorithm in data poisoning attacks.

SGD is a more basic gradient descent method. In this experiment, we chose SGD as compared algorithm. This experiment compares the performance of SGD, Adam, and SSGD on the same data set with noises. To determine the scale of added noises, the differential privacy mechanisms still are used to add noises with the same methods as the robustness experiment. We add Laplacian noises of different scales to 60,000 training samples. The evaluation method of the experiment result is the same as the above experiment. On MNIST, the batch number is set to $n = 64$ and $m = 4$. The learning rate of SGD, Adam, and SSGD is 10^{-4} , 10^{-4} , and 10^{-2} , respectively. On Fashion-MNIST, the batch number is set to $n = 64$ and $m = 16$. The learning rate of SGD, Adam, and SSGD is 10^{-4} , 10^{-4} , and $10^{-2}/1.5^{0.002j}$, respectively. The other settings are the same as the above experiment.

Figure 3 is the effect picture after adding different noise scales. From Tables 9 and 10, we can see that SSGD is significantly better than SGD and Adam in test accuracy. Also, our algorithm still maintains a higher test accuracy while continuously increasing the scale of noises. Therefore,

SSGD can resist gradient poisoning attacks and parametric poisoning attacks to a certain extent.

4.4. Data Reconstruction Attack. Zhu et al. [7] presented an approach which shows the possibility of obtaining private training data from the publicly shared gradients. In their deep leakage from gradient (DLG) method, they synthesized the dummy data and corresponding labels with the supervision of shared gradients. Specifically, they start with random initialization of pseudodata and labels. Virtual gradients are computed on the current shared model in the distributed setup. By minimizing the difference between the virtual gradient and the shared real gradient, they iteratively update the virtual data and labels simultaneously. iDLG [4] is an improvement based on DLG. The following experimental diagrams include the experimental results of DLG [7] and iDLG [4] attacking SGD and SSGD algorithms on MNIST datasets and Fashion-MNIST datasets. Figures 4 and 5 are the experimental results of DLG and iDLG attacking SGD. Figures 6 and 7 are about the experimental results of DLG and iDLG attacking SSGD. The number of iterations is 300, and the iteration is stopped if the

predetermined accuracy is reached. We can see that our algorithm can defend against DLG and iDLG.

5. Conclusions

In this paper, we propose a new gradient descent approach, called super stochastic gradient descent. The SSGD enhances the randomness of gradients to protect against gradient-based attacks. Simultaneously, we use multisample aggregation to enhance stability and eliminate the uncertainty brought about by superrandomness. Our approach achieves neuron-level security and can defend against attacks on the gradient. Experimental results demonstrate that SSGD has good accuracy and strong robustness because its stability and randomness are enhanced. SSGD can also resist model poisoning attacks to a certain extent. But for attacks with the same degree of poisoning, data poisoning has a greater impact on performance. In the future, we will continue to find a more suitable method for resisting data poisoning attacks.

Data Availability

All the experimental data used to support the findings of this study are included within the article.

Disclosure

An earlier version of this study's preprint is given in the following link: <https://arxiv.org/abs/2012.02076>.

Conflicts of Interest

The authors declare that they have no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (nos. 61672176 and 61763003), Research Fund of Guangxi Key Lab of Multi-Source Information Mining and Security (no. 19-A-02-01), Guangxi 1000-Plan of Training Middle-Aged/Young Teachers in Higher Education Institutions, Guangxi "Bagui Scholar" Teams for Innovation and Research Project, Guangxi Talent Highland Project of Big Data Intelligence and Application, and Guangxi Collaborative Innovation Center of Multi-source Information Integration and Intelligent Processing.

References

- [1] S. Ruder, "An overview of gradient descent optimization algorithms," 2016, <https://arxiv.org/abs/1609.04747>.
- [2] L. Yang and D. Cai, "AdaDB: an adaptive gradient method with data-dependent bound," *Neurocomputing*, vol. 419, pp. 183–189, 2021.
- [3] L. Melis, C. Song, E. D. Cristofaro, and V. Shmatikov, "Exploiting unintended feature leakage in collaborative learning," in *Proceedings of the IEEE Symposium on Security and Privacy (SP)*, pp. 691–706, San Francisco, CA, USA, May 2019.
- [4] B. Zhao, K. R. Mopuri, and H. Bilen, "iDLG: improved deep leakage from gradients," 2020, <https://arxiv.org/abs/2001.02610>.
- [5] B. Hitaj, G. Ateniese, and F. Pérez-Cruz, "Deep models under the GAN: Information leakage from collaborative deep learning," in *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security (CCS)*, pp. 603–618, Dallas, TX, USA, November 2017.
- [6] Z. Wang, M. Song, Z. Zhang, Y. Song, Q. Wang, and H. Qi, "Beyond inferring class representatives: user-level privacy leakage from federated learning," in *Proceedings of the IEEE Conference on Computer Communications (INFOCOM)*, pp. 2512–2520, Paris, France, April 2019.
- [7] L. Zhu, Z. Liu, and S. Han, "Deep leakage from gradients," in *Proceedings of the Annual Conference on Neural Information Processing Systems 2019 (NeurIPS)*, pp. 14747–14756, Vancouver, Canada, December 2019.
- [8] J. Geiping, H. Bauermeister, H. Droge, and M. Moeller, "Inverting gradients-how easy is it to break privacy in federated learning?" in *Proceedings of the Annual Conference on Neural Information Processing Systems 2020 (NeurIPS) Virtual Event*, December 2020.
- [9] X. Pan, M. Zhang, Y. Yan, J. Zhu, and M. Yang, "Theory-oriented deep leakage from gradients via linear equation solver," 2020, <https://arxiv.org/abs/2010.13356>.
- [10] 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 (CVPR)*, pp. 770–778, Las Vegas, NV, USA, June 2016.
- [11] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proceedings of the International Conference on Learning Representations (ICLR)*, San Diego, CA, USA, May 2015.
- [12] G. Huang, Z. Liu, L. V. D. Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2261–2269, Honolulu, HI, USA, July 2017.
- [13] A. Krizhevsky, "One weird trick for parallelizing convolutional neural networks," 2014, <https://arxiv.org/abs/1404.5997>.
- [14] N. Ma, X. Zhang, H. Zheng, and J. Sun, "Shufflenet V2: practical guidelines for efficient CNN architecture design," in *Proceedings of the 15th European Conference on Computer Vision (ECCV)*, pp. 122–138, Munich, Germany, September 2018.
- [15] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2818–2826, Las Vegas, NV, USA, June 2016.
- [16] C. Szegedy, W. Liu, Y. Jia et al., "Going deeper with convolutions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1–9, Boston, MA, USA, June 2015.
- [17] M. Sandler, A. G. Howard, M. Zhu, A. Zhmoginov, and L. Chen, "Mobilenetv2: inverted residuals and linear bottlenecks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4510–4520, Salt Lake City, UT, USA, June 2018.
- [18] K. Bonawitz, V. Ivanov, B. Kreuter et al., "Practical secure aggregation for privacy-preserving machine learning," in *Proceedings of the ACM SIGSAC Conference on Computer*

- and Communications Security (CCS), pp. 1175–1191, Dallas, TX, USA, October 2017.
- [19] G. Xu, H. Li, S. Liu, K. Yang, and X. Lin, “VerifyNet: secure and verifiable federated learning,” *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 911–926, 2020.
 - [20] X. Guo, Z. Liu, J. Li et al., “VeriFL: communication-efficient and fast verifiable aggregation for federated learning,” *IEEE Transactions on Information Forensics and Security*, vol. 16, pp. 1736–1751, 2021.
 - [21] J. H. Bell, K. A. Bonawitz, A. Gascón, T. Lepoint, and M. Raykova, “Secure single-server aggregation with (Poly) logarithmic overhead,” in *Proceedings of the ACM SIGSAC Conference on Computer and communications security (CCS)*, pp. 1253–1269, Virtual Event, USA, November 2020.
 - [22] B. Choi, J.-y. Sohn, D.-J. Han, and J. Moon, “Communication computation efficient secure aggregation for federated learning,” 2020, <https://arxiv.org/abs/2012.05433>.
 - [23] H. Fereidooni, S. Marchal, M. Miettinen et al., “SAFELearn: Secure aggregation for private FEDerated learning,” in *Proceedings of the 2021 IEEE Security and Privacy Workshops (SPW)*, pp. 56–62, Virtual Event, May 2021.
 - [24] X. Ma, C. Ji, X. Zhang et al., “Secure multiparty learning from the aggregation of locally trained models,” *Journal of Network and Computer Applications*, vol. 167, Article ID 102754, 2020.
 - [25] L. T. Phong, Y. Aono, T. Hayashi, L. Wang, and S. Moriai, “Privacy-Preserving deep learning via additively homomorphic encryption,” *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 5, pp. 1333–1345, 2018.
 - [26] M. Abadi, A. Chu, I. J. Goodfellow et al., “Deep Learning with Differential Privacy,” in *Proceedings Of the ACM SIGSAC Conference on Computer and communications security (CCS)*, pp. 308–318, Vienna, Austria, October 2016.
 - [27] K. Yadav, B. B. Gupta, K. T. Chui, and K. E. Psannis, “Differential privacy approach to solve gradient leakage attack in a federated machine learning environment,” in *Proceedings of the International Conference on Computational Data and Social Network (CSoNet)*, pp. 378–385, Dallas, TX, USA, December 2020.
 - [28] Q. Zhao, C. Zhao, S. Cui, S. Jing, and Z. Chen, “PrivateDL: privacy-preserving collaborative deep learning against leakage from gradient sharing,” *International Journal of Intelligent Systems*, vol. 35, no. 8, pp. 1262–1279, 2020.
 - [29] K. Kawaguchi, “Deep learning without poor local minima,” in *Proceedings of the Annual Conference on Neural Information Processing Systems (NeurIPS)*, pp. 586–594, Barcelona, Spain, May 2016.
 - [30] H. Xiao, K. Rasul, and R. Vollgraf, “Fashion-MNIST: a novel image dataset for benchmarking machine learning algorithms,” 2017, <https://arxiv.org/abs/1708.07747>.
 - [31] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.
 - [32] N. Qian, “On the momentum term in gradient descent learning algorithms,” *Neural Networks*, vol. 12, no. 1, pp. 145–151, 1999.
 - [33] D. P. Kingma and J. Ba, “Adam: a method for stochastic optimization,” in *Proceedings of the International Conference On Learning Representations (ICLR)*, San Diego, CA, USA, May 2015.

Research Article

A Lightweight and Secure Anonymous User Authentication Protocol for Wireless Body Area Networks

Junsong Zhang ¹, Qikun Zhang,¹ Zhigang Li,¹ Xianling Lu ² and Yong Gan³

¹School of Computer and Communication Engineering, Zhengzhou University of Light Industry, Zhengzhou 450002, China

²School of Information Engineering, Zhengzhou University of Industrial Technology, Zhengzhou 450002, China

³School of Information Engineering, Zhengzhou University of Technology, Zhengzhou 450002, China

Correspondence should be addressed to Xianling Lu; 2014102@zzuli.edu.cn

Received 19 May 2021; Accepted 6 July 2021; Published 22 July 2021

Academic Editor: Jie Cui

Copyright © 2021 Junsong Zhang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The recent development of wireless body area network (WBAN) technology plays a significant role in the modern healthcare system for patient health monitoring. However, owing to the open nature of the wireless channel and the sensitivity of the transmitted messages, the data security and privacy threats in WBAN have been widely discussed and must be solved. In recent years, many authentication protocols had been proposed to provide security and privacy protection in WBANs. However, many of these schemes are not computationally efficient in the authentication process. Inspired by these studies, a lightweight and secure anonymous authentication protocol is presented to provide data security and privacy for WBANs. The proposed scheme adopts a random value and hash function to provide user anonymity. Besides, the proposed protocol can provide user authentication without a trusted third party, which makes the proposed scheme have no computational bottleneck in terms of architecture. Finally, the security and performance analyses demonstrate that the proposed scheme can meet security requirements with low computational and communication costs.

1. Introduction

In recent years, along with the quick development of communications and microelectronics technologies, a new network paradigm for detecting human body data, named wireless body area networks (WBANs) [1], has emerged. A typical architecture of WBAN for the healthcare system is depicted in Figure 1. There are three main participants in the WBAN: a dynamic set of M patients with monitoring sensors, denoted as $PAT = \{P_j | j = 1, 2, \dots, M\}$, a set of N doctors as $DCT = \{D_i | i = 1, 2, \dots, N\}$, and a registration center (RC) as a trusted third party [2]. The sensors are mainly embedded or worn on the patient. Their main function is to collect various physical parameters of the patient, such as blood pressure (BP), electrocardiogram (ECG), and temperature, and then transmit these data to the personal terminal. Next, the personal terminal uses a wireless communication technology (such as Wi-Fi and 4G/5G/CDMA) to forward all collected information to the appropriate

doctor or the medical server. Therefore, the personal terminal acts as a bridge between the doctors and WBAN. These sensory data collected from the patient will play an important role in the doctor's medical diagnosis. In addition, this new technology not only helps to monitor and improve the health of patients but is also more suitable for health monitoring and care for the elderly and the disabled. However, due to the openness of the wireless channel, the data transmitted in WBAN can easily be eavesdropped or tampered with by unauthorized users. Since these sensitive patient data are the basis of clinical diagnosis, any data leakage or modification may put the patient's life at risk [3–5]. Consequently, it is necessary and important to provide a safe and reliable authentication protocol in the WBAN to ensure that only legitimate users can obtain the patient's sensitive information.

Since the collected information is vital to the patient's life, it is very confidential and vulnerable to various attacks by an adversary. If these sensitive data are obtained and

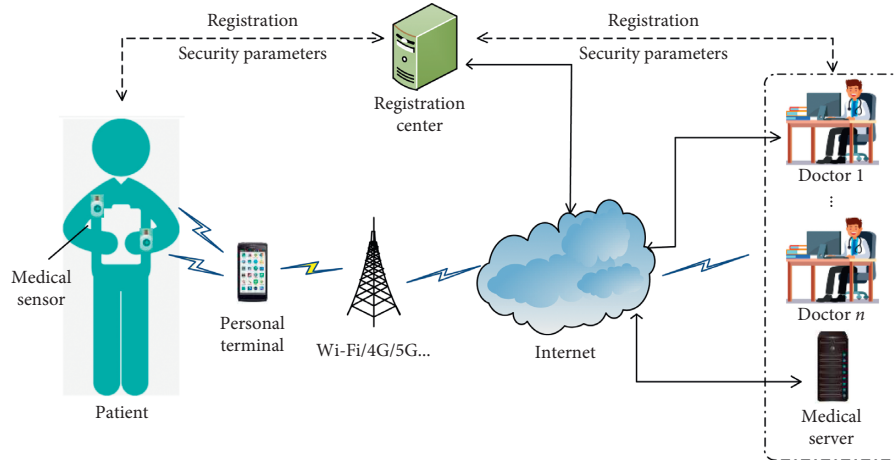


FIGURE 1: A typical system model of the WBAN.

misused by an adversary, it may threaten the lives of patients. Therefore, it is important to provide data security and privacy protection to the WBAN [6]. In other words, strong security solutions and authentication protocols are necessary for the success and large-scale deployment of the WBANs. Motivated by these shortcomings, we proposed a lightweight and secure anonymous user authentication protocol for the WBAN. The contributions of the paper are summarized as follows:

- (1) To guarantee the privacy of doctors and patients in the WBAN, an efficient ECC-based privacy-preserving authentication is proposed. Moreover, the proposed authentication protocol can verify the legitimacy of the patients and doctors.
- (2) In the proposed authentication protocol, under the premise of anonymous authentication of users, no trusted third party is required to participate in the authentication process. In this way, the proposed authentication protocol has no computational bottleneck in terms of architecture. Besides, the proposed scheme can provide a low computation burden on the client side, which makes the proposed authentication protocol more efficient.
- (3) The proposed authentication protocol provides a method for RC to track the doctor's actual identity. At the same time, it also ensures that the doctor's identity information is not obtained by unrelated parties. This makes it possible to prevent doctors from making a wrong diagnosis or to pursue accountability afterward.
- (4) A detailed security analysis and performance analysis show that the proposed authentication protocol can meet the security and performance requirements of the WBAN application.

The rest of the paper is organized as follows. Section 2 discusses the existing secure authentication schemes. Section 3 describes the attacker models and preliminaries. Section 4 presents the proposed mutual authentication scheme. Security and performance analyses of the proposed protocol

are provided in Sections 5 and 6, respectively. Finally, Section 7 gives the conclusion of this paper.

2. Related Work

Security, privacy, and identity authentication are the most critical and challenging issues in the WBAN. During the last few years, so many authentication protocols have been proposed to solve the security and privacy protection problem for wireless-based healthcare applications [7–10]. Some research activities use public key cryptography (PKC) to build authentication schemes [7, 8]. Since the traditional PKC requires a large amount of computation overhead, these existing PKC-based methods are not suitable for the resource-constrained WBAN. In 2014, Chatterjee et al. [9] presented an ECC-based user authentication for WBAN. Liu et al. [10] proposed a lightweight certificateless authentication scheme that uses ECC and bilinear pairings. Unfortunately, their method was found to be unable to resist tracking attack and impersonation attack [11].

In 2015, Das et al. [12] suggested a biometric-based authentication protocol for WBAN. Their proposed protocol combines biometric information and a password to verify the legitimacy of the user. Later, Wang and Zhang [13] found that Das et al.'s scheme is not able to provide user anonymity. In order to avoid this defect, they proposed a new bilinear pairing-based authentication protocol in the WBAN environment. In the same year, Debiao et al. [14] presented a bilinear pairing-based anonymous authentication scheme for WBAN. Liu et al. [15] proposed an anonymous 1-round authentication protocol for WBANs. They claimed that their authentication scheme was efficient and secure. However, Li et al. [16] demonstrated that Liu et al.'s scheme is unable to resist impersonation attack, DoS attack, and session key guessing attack. To avoid these flaws, they proposed an improved 1-round authentication protocol for WBANs. Later, Shen et al. [17] presented a lightweight nonpairing certificateless authentication protocol for WBANs. Unfortunately, their proposed scheme was found to be unable to resist the impersonation attack. To remove the flaws, Liu

et al. [18] proposed an improved authentication to remedy the flaws in Shen et al.'s scheme. Wazid et al. [19] proposed a novel authentication and key management scheme for the cloud-assisted WBAN.

Later, Qiu et al. [20] proposed a secure mutual authentication protocol based on ECC for wireless medical sensor networks. In this paper, the BAN logic is used to prove the security of the proposed scheme. However, according to [21], it is still suffering from insider attack. Shen et al. [21] presented a cloud-aided certificateless and privacy-preserving authentication scheme for the WBAN. In [21], the authors use public key cryptography and the message authentication code (MAC) to achieve user authentication. Shuai et al. [22] presented a bilinear pairing-based mutual authentication scheme for WBAN. Fotouhi et al. [23] propose a new lightweight hash chain-based and forward secure authentication scheme for WBAN. Kumar et al. [24] presented an ECC-based authentication scheme for wearable devices environment. Jegadeesan et al. [25] proposed an efficient privacy-preserving anonymous authentication for WBAN. However, their scheme is also not able to resist the impersonation attack.

To enhance the security of WBAN, a novel lightweight and secure anonymous user authentication protocol was designed. Compared with other existing schemes, the scheme proposed in this paper has two distinct characteristics. First, the proposed scheme does not require a trusted third party to verify the legitimacy of users anonymously. Second, the proposed authentication protocol provides a method for RC to track the doctor's actual identity, which can reduce the doctor's misbehaving.

3. Preliminaries

3.1. Threat Model. An adversary model is a valid abstraction of an arbitrary adversary which is able to launch a successful attack. Due to the open nature of WBAN, the wireless communication channel is vulnerable to various attacks. In the proposed authentication protocol, the two widely used models, named Dolev-Yao model and CK-adversary model, are used. In the Dolev-Yao model, the communication between different entities can be intercepted by an adversary. Besides, the adversary is also able to modify/delete/fake/inject into the transmitting information [26, 27]. In the CK-adversary model, the adversary can control all the communication between the entities. Moreover, the adversary is assumed to be able to extract the secret parameters stored in the entity's memory and the temporary data used to establish session keys [6]. Furthermore, the adversary can use oracle queries to interact with the entities. As far as we know, these two adversary models are widely adopted in the authenticated key exchange protocols [28].

3.2. Security Requirements for the WBAN. The communication of the WBAN is mainly divided into two types: the communication between the sensor and the personal terminal and the communication between the personal terminal and the back-end server. Our work focuses on the

security of communication between the personal terminal and the back-end server. In this section, we discuss the security and privacy requirements for the WBAN environment [29].

3.2.1. Mutual Authentication. As we all know, the messages transmitted in the WBAN are easily eavesdropped and modified. Hence, once a message is received, the most important thing for the receiver is to determine whether the message is sent by a legitimate user and whether the message has been modified. Therefore, there should be a mechanism to verify the legitimacy of the message and the sender of the message.

3.2.2. Data Integrity. To ensure the integrity of the transmitted message in the WBAN, an anonymous signature mechanism is attached to the transmitted message.

3.2.3. Confidentiality. Since the messages transmitted in the WBAN contain the patient's sensitive information, and this sensitive information is very important privacy for patients. Therefore, the proposed protocol needs to ensure that the unauthorized entities cannot obtain the content of the transmitted message.

3.2.4. Identity Privacy-Preserving. To protect the identity privacy of users (especially the patients), the actual identity of the patients cannot directly appear in the transmitted messages. Besides, the proposed protocol also needs to ensure that the adversary cannot decipher/calculate the patient's actual identity through the message.

3.2.5. Conditional Traceability. In WBAN, for the manager, the doctor's identity should be traceable. Especially when a doctor makes any dispute or misbehavior, the manager needs to have the ability to get the doctor's actual identity. This provides a basis for subsequent accountability and can also reduce the loss of WBAN.

3.2.6. Attack Resistance. To ensure secure communication in WBANs, the proposed protocol should be able to withstand various common attacks, such as replay attack, impersonation attack, and man-in-the-middle attack.

3.3. Elliptic Curve Cryptography. Elliptic curve cryptography (ECC) is one of the most widely used public key asymmetric cryptographies [30]. Its security comes from the discrete logarithm problem (DLP) in a group defined by points on elliptic curve. An elliptic curve E over $GF(p)$, where p is a large prime, is defined by an equation of the following form:

$$y^2 = x^3 + ax + b, \quad (1)$$

where $a, b \in GF(p)$ and satisfies $4a^3 + 27b^2 \neq 0 \pmod{p}$. There are two basic operations on ECC: point addition and

scalar multiplication. The scalar multiplication over E can be computed by repeated addition as

$$k \cdot P = P + P + \dots + P \text{ (} k \text{ times)}. \quad (2)$$

The hardness of the elliptic curve discrete logarithm problem is essential for the security of all elliptic curve cryptographic schemes. Here, we present two important mathematical problems on elliptic curves as follows [31]:

Elliptic curve discrete logarithm problem (ECDLP): given an elliptic curve E defined over a finite field $GF(p)$, and two points $Q, P \in E$ of order q , it is hard to find an integer $k \in \mathbb{Z}^*q$ such that $Q = k \cdot P$

Elliptic curve Diffie–Hellman problem (ECDHP): given an elliptic curve E defined over a finite field $GF(p)$, a point $P \in E$ of order n , $A = aP$, $B = bP$, and find the point $C = abP$

4. The Proposed Authentication Protocol

In this section, we present our proposed authentication protocol for WBAN. The proposed protocol consists of three phases: system initialization, registration, and anonymous mutual authentication. All the notations used in this paper are presented in Table 1. The detailed descriptions of these phases are explained as follows.

4.1. System Initialization. In the proposed authentication protocol, as mentioned earlier, RC is considered as a trusted third party. It is responsible for the registration of all patients and doctors in the WBAN. At the same time, it must also set relevant security parameters for the authentication protocol.

Step I-1: RC selects an appropriate elliptic curve E over the finite field $GF(p)$. Then, RC chooses a bilinear mapping $\hat{e}: G_1 \times G_1 \rightarrow G_2$ and the generator $P_0 \in G_1$ with the order q over elliptic curve E , where q is a big prime number.

Step I-2: RC chooses two secure hash function h and H , where $h: \{0, 1\}^* \rightarrow \mathbb{Z}^*q$, $H: \{P \in E\} \rightarrow \{0, 1\}^l$, in which l is the length of the string. Next, RC selects two random number $u, v \in \mathbb{Z}^*q$ as secret values and keeps them properly.

Step I-3: RC chooses a random number s_{RC} as its master key and computes the corresponding public key $PK_{RC} = s_{RC} \cdot P$. Then, RC publishes the public system parameters to the users: $param = \{E, G_1, G_2, PK_{RC}, h, H, \hat{e}\}$.

4.2. Registration. This phase consists of the doctor registration and the patient registration. The process of registration is explained as follows:

Doctor registration: when a doctor D_i wants to login to the system to get the patient's information, he/she must first register at RC through the following steps:

Step DR-1: the doctor D_i chooses his/her own identification DID_i and password DPW_i and a random

number r_i and then computes $h(r_i \oplus DPW_i)$. Then, D_i sends the message $\{DID_i, h(r_i \oplus DPW_i)\}$ to RC via a secure channel.

Step DR-2: upon receiving the message $\{DID_i, h(r_i \oplus DPW_i)\}$, RC computes $A_i = h(DI D_i | v)$, $B_i = h(A_i)$, $V_i = A_i \oplus h(DID_i || h(r_i \oplus DPW_i))$. Then, RC regards the parameter $s_{D_i} = h(r_i \oplus DPW_i)$ as the doctor D_i 's master key and then computes the corresponding public key $PK_{D_i} = s_{D_i} \cdot P$.

Step DR-3: RC provides a license to the doctor D_i : $L_{D_i} = s_{D_i} \cdot v \cdot P$, then RC maintains $\langle DID_i, L_{D_i} \rangle$ in the checklist. This checklist is used to check the actual identity of the doctor when the doctor makes any dispute or misbehavior.

Step DR-4: the RC issues a smart card to the doctor D_i , the card contains the values $\{B_i, V_i, PK_{D_i}, L_{D_i}, r_i\}$. After receiving the smart card, the doctor D_i inserts the value r_i into the smart card. Then, the smart card contains $\{B_i, V_i, PK_{D_i}, L_{D_i}, r_i\}$.

Patient registration: when the patient P_j is ready to go to the hospital for treatment, RC will register his/her handheld terminal and assign relevant medical sensors to him/her to monitor the physical parameters.

Step PR-1: RC chooses a random number $s_{P_j} \in \mathbb{Z}^*p$ as the patient P_j 's master key. And then RC computes the corresponding public key $PK_{P_j} = s_{P_j} \cdot P$. Next, RC sends the message $\{s_{P_j}, PK_{P_j}\}$ to the patient P_j through a secure channel.

4.3. Anonymous Authentication

4.3.1. Patient to Doctor Anonymous Authentication. When the patient P_j wants to send the data collected by himself to the doctor D_i to facilitate the doctor's diagnosis or detection, this step is required. Since the data transmitted by the patient to the doctor contain very sensitive health information, in order to preserve the privacy of these data, the patient needs to use encryption and authentication methods to process the data. The detailed steps are as follows:

Step PA-1: the patient P_j first chooses a random value $k \in \mathbb{Z}^*p$ and calculates

$$\begin{aligned} a_1 &= k \cdot P, \\ a_2 &= k \cdot PK_{D_i}, \\ a_3 &= h(\text{data})k \cdot s_{P_j} \cdot PK_{RC}, \\ a_4 &= k \cdot PK_{P_j}, \\ w_1 &= (\text{data}a3a4Tj), \\ c_1 &= w1 \oplus H(a_2), \end{aligned} \quad (3)$$

where data are the physical parameters of the patient P_j and T_j is the timestamp. Then, the patient P_j sends the message $\{a_1, c_1, T_j\}$ to the doctor D_i via common channel.

Step PA-2: upon receiving the message $\{a_1, c_1, T_j\}$, the doctor D_i computes $w^*1 = c_1 \oplus H(s_{D_i}a_1)$ and extracts

TABLE 1: Notation and its description.

Notation	Description
D_i	The i th doctor
DID_i	The identity of the i th doctor
PK_{D_i}	The i th doctor's public key
RC	The registration center
PK_{RC}	The public key of RC
P_j	The j th patient
PID_j	The identity of the j th patient
$h(\cdot)$	A secure hash function, where $h: \{0, 1\}^* \Rightarrow Z_q^*$
$H(\cdot)$	A hash function, where $H: E_p(a, b) \Rightarrow \{0, 1\}^l$, in which l is the length of the string
$\hat{e}(\cdot, \cdot)$	A bilinear map $\hat{e}: G_1 \times G_1 \longrightarrow G_2$
\parallel	String concatenation operation
\oplus	The bitwise XOR operation

the $data$, a_3 , a_4 and the timestamp T_j from w^*1 . Then, the doctor D_i verifies whether the timestamp T_j is fresh. If it is not fresh, the doctor D_i discards the message directly and terminates the authentication process. Otherwise, go to the next step.

Step PA-3: the doctor D_i checks if $\hat{e}(a_3, PK_{D_i})? = \hat{e}(PK_{RC}, h(data) \cdot s_{D_i} \cdot a_4)$ holds. If the above equation is true, the doctor D_i considers that the patient P_j is legitimate and the health information $data$ have not been destroyed. Otherwise, the patient P_j is considered to be an illegal user and refuses to accept the health information data.

Figure 2 summarizes the process of patient to doctor authentication phase.

Proof of Correctness. The challenger equation $\hat{e}(a_3, PK_{D_i})? = \hat{e}(PK_{RC}, h(data) \cdot s_{D_i} \cdot a_4)$ calculated by the doctor D_i should be held by using the values a_3 and a_4 sent from the patient P_j .

$$\begin{aligned}
\hat{e}(a_3, PK_{D_i}) &= \hat{e}(h(data) \cdot k \cdot s_{P_j} \cdot PK_{RC}, PK_{D_i}) \\
&= \hat{e}(k \cdot s_{P_j} \cdot PK_{RC}, h(data) \cdot s_{D_i} \cdot P) \\
&= \hat{e}(PK_{RC}, h(data) \cdot k \cdot s_{P_j} \cdot s_{D_i} \cdot P) \quad (4) \\
&= \hat{e}(PK_{RC}, h(data) \cdot s_{D_i} \cdot k \cdot PK_{P_j}) \\
&= \hat{e}(PK_{RC}, h(data) s_{D_i} \cdot a_4).
\end{aligned}$$

4.3.2. Doctor to Patient Anonymous Authentication. When the doctor D_i wants to get the relevant health data of the patient P_j , he first generates the query information $demand$ and completes the message authentication through the following steps:

Step DA-1: the doctor D_i first inserts his/her smart card to a terminal and then inputs his/her identity DID_i and password DPW_i . Then, the smart card computes as follows: $A_i^* = h(DI D_i h(r_i \oplus DPW_i)) \oplus V_i$, $B_i^* = h(A_i^*)$, and checks whether $B_i^* = B_i$. If not, the smart card rejects this request and prompts

the doctor to enter the correct identity and password. Otherwise, go to the next step.

Step DA-2: the doctor D_i chooses a random number $r \in Z^*p$ and computes

$$\begin{aligned}
b_1 &= r \cdot P, \\
b_2 &= r \cdot PK_{P_j}, \\
b_3 &= h(demand) r \cdot s_{D_i} \cdot PK_{RC}, \\
b_4 &= r \cdot PK_{D_i}, \\
b_5 &= h(T_i) \cdot s_{D_i} \cdot P, \\
Cert_i &= (L_{D_i} T_i) \oplus H(h(T_i) \cdot s_{D_i} \cdot PK_{RC}), \\
w_2 &= (demand b_3 b_4 Cert_i T_i), \\
c_2 &= w_2 \oplus H(b_2),
\end{aligned} \quad (5)$$

where $demand$ is the query request information of the doctor and T_i is the timestamp. Then, the doctor D_i sends the message $\{b_1, b_5, c_2, Cert_i, T_i\}$ to the patient P_j via a common channel.

Step DA-3: upon receiving the message $\{b_1, b_5, c_2, Cert_i, T_i\}$, the patient P_j verifies whether the time stamp T_i is fresh. If not, the authentication process is terminated. Otherwise, P_j uses his/her private key to compute $b_2^* = s_{P_j} \cdot b_1$, $w_2^* = c_2 \oplus H(b_2^*)$. And then, P_j extracts variables $demand$, b_3 , b_4 , $Cert_i$ and the timestamp T_i from w^*2 .

Step DA-4: P_j verifies whether the equation $\hat{e}(b_3, PK_{P_j})? = \hat{e}(PK_{RC}, h(demand) \cdot s_{P_j} \cdot b_4)$ holds. If the above equation is true, the patient P_j considers the doctor to be a legitimate doctor, and he will provide the relevant health data according to the doctor's requirements. Otherwise, he believes that the doctor D_i is an illegal doctor and refuses to accept his request.

Figure 3 summarizes the process of login and the doctor to patient authentication phase.

Proof of correctness:

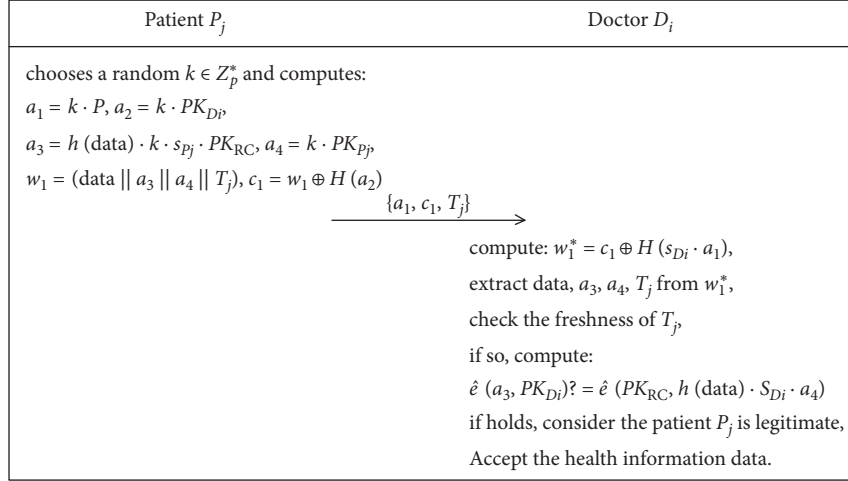


FIGURE 2: The patient to doctor authentication phase.

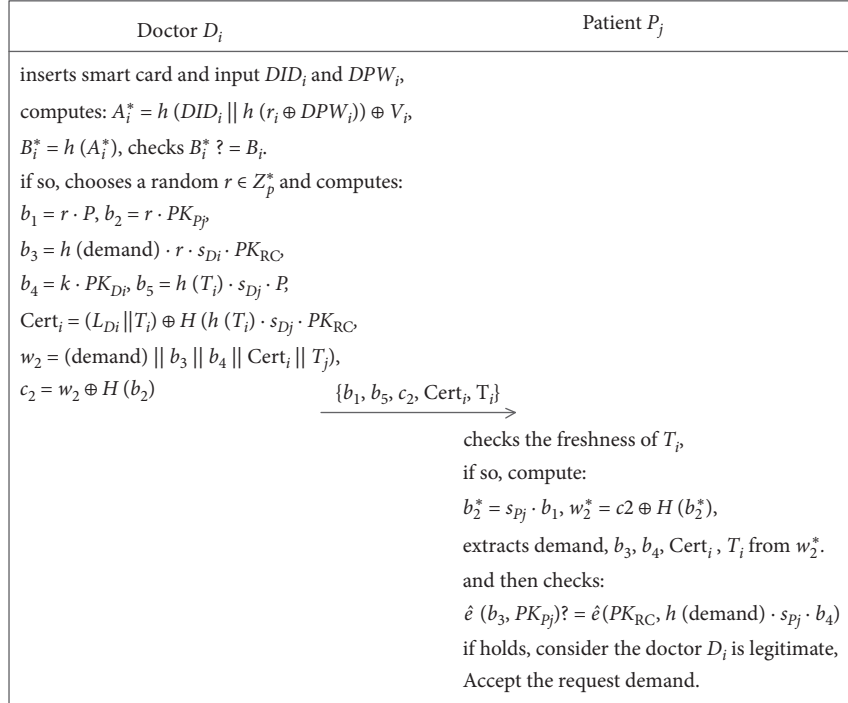


FIGURE 3: The login process and doctor to patient authentication phase.

$$\begin{aligned}
 \hat{e}(b_3, PK_{P_j}) &= \hat{e}(h(\text{demand}) \cdot r \cdot s_{D_i} \cdot PK_{RC}, s_{P_j} \cdot P) \\
 &= \hat{e}(PK_{RC}, h(\text{demand}) \cdot r \cdot s_{D_i} \cdot s_{P_j} \cdot P) \\
 &= \hat{e}(PK_{RC}, h(\text{demand}) \cdot s_{P_j} \cdot r \cdot s_{D_i} \cdot P) \\
 &= \hat{e}(PK_{RC}, h(\text{demand}) \cdot s_{P_j} \cdot b_4).
 \end{aligned} \tag{6}$$

Step DA-5 (*identity tracking*): if the request message *demand* is suspected of having a problem or illegal operation, RC is able to track the actual identity of the

doctor using the certificate $Cert_i$ in the message. The process is as follows:

Then, RC finds the corresponding record $\langle DID_i, L_{D_i} \rangle$ in the checklist and gets the actual identity of the doctor DID_i .

$$\begin{aligned}
 &Cert_i \oplus H(s_{RC} \cdot b_5) \\
 &= (L_{D_i} T_i) \oplus H(h(T_i) \cdot s_{D_i} \cdot PK_{RC}) \oplus H(s_{RC} \cdot b_5) \\
 &= (L_{D_i} T_i) \oplus H(h(T_i) \cdot s_{D_i} \cdot PK_{RC}) \oplus H(s_{RC} \cdot h(T_i) \cdot s_{D_i} \cdot P) \\
 &= (L_{D_i} T_i).
 \end{aligned} \tag{7}$$

5. Security Analysis

In this section, we first prove that the proposed anonymous user authentication protocol is provably secure under the BAN logic [32, 33]. Next, the security and functional features of the proposed authentication protocol are discussed.

5.1. BAN Logic-Based Formal Security Analysis. We use BAN logic to analyze the security and correctness of our proposed authentication protocol. Table 2 summarizes the notations and rules of the BAN logic.

Goals. According to the analytic procedures of the BAN logic, the proposed authentication protocol must satisfy the following security goals:

$$\text{Goal}_1: P_j | \equiv D_i | \equiv P_j \xleftrightarrow{c_1} D_i$$

$$\text{Goal}_2: D_i | \equiv P_j | \equiv D_i \xleftrightarrow{c_2} P_j$$

The initial status forms of the proposed authentication protocol are formally described as follows:

$$A_1: D_i | \equiv \#(T_i, r)$$

$$A_2: P_j | \equiv \#(T_j, k)$$

$$A_3: D_i \triangleleft \{a_3, \text{data}\}_{H(a_2)}$$

$$A_4: P_j \triangleleft \{\text{demand}, T_i\}_{H(b_2)}$$

The idealized transformed message of the proposed authentication protocol is described as follows:

$$\text{Msg}_1: P_j \longrightarrow D_i: \{a_1, c_1, T_j\}$$

$$\text{Msg}_2: D_i \longrightarrow P_j: \{b_1, b_5, c_2, \text{Cert}_i, T_i\}$$

The main analysis steps of the proposed authentication protocol based on the BAN logic are described as follows:

By A_2 , A_3 , and the message meaning rule, it is easy to get $S_1: D_i | \equiv P_j | \sim \{a_3, \text{data}\}_{H(a_2)}$

By S_1 , A_3 , Msg_1 , and the nonce verification rule in which k is the necessary part of $H(a_2)$, it is easy to get $S_2:$

$$P_j | \equiv D_i | \equiv P_j \xleftrightarrow{H(a_2)} D_i$$

By S_2 , Msg_1 , and the nonce verification rule in which T_j is the part of c_1 , it is easy to get $S_3: P_j | \equiv D_i | \equiv P_j \xleftrightarrow{c_1} D_i$ (Goal_1)

By A_1 , A_4 , and the message meaning rule, it is easy to get $S_4: P_j | \equiv D_i | \sim \{b_5, c_2, T_i\}_{H(b_2)}$

By S_4 , A_4 , Msg_1 , and the nonce verification rule in which r is the necessary part of $H(b_2)$, it is easy to get $S_5:$

$$D_i | \equiv P_j | \equiv D_i \xleftrightarrow{H(b_2)} P_j$$

By S_5 , Msg_2 and the nonce verification rule in which T_i is the part of c_2 , it is easy to get $S_6: D_i | \equiv P_j | \equiv D_i \xleftrightarrow{c_2} P_j$ (Goal_2)

5.2. Informal Security Analysis. In this section, the security and functional features of the proposed authentication protocol are discussed. Through the detailed analysis, it has been proven that the proposed protocol can withstand various common attacks.

5.2.1. Privileged Insider Attack. In the proposed protocol, RC does not store any patient-related information. Therefore, the privileged insider cannot obtain any critical information about the patient. In another, although RC stores the doctor's checklist $\langle DID_i, L_{Di} \rangle$ to track the doctor's true identity, the privileged insider cannot guess the doctor's password DPW_i or private key s_{Di} . Therefore, he/she has no advantage in breaking the robustness of the proposed authentication protocol.

5.2.2. Replay Attack. Owing to the open nature of the wireless communication channel, the replay attack poses a great security threat to the wireless body area networks. According to the specification of the proposed protocol, the first step of each entity (the patient or doctor) is to check the freshness of the authentication messages using the timestamps T_i or T_j . In addition, the timestamp is hashed and Exclusive OR (\oplus) with other parameters (c_1 , c_2 , or b_5), which is contained in the authentication messages. Therefore, if the timestamp is not fresh, the receiver discards the message directly and aborts the session. If the adversary modifies the timestamp, he/she cannot calculate the corresponding parameters. Consequently, our proposed protocol is able to withstand the replay attack.

5.2.3. Impersonation Attack. Let A be an adversary and he has the ability to intercept the authentication message of the patient $P_j: \{a_1, c_1, T_j\}$. A may try to generate a forged authentication message $\{a^*1, c^*1, T^*1\}$. Since A has not registered at RC and does not know the secret value u , it is impossible for A to obtain its own correct public key PK^*P_j . Even though the adversary A chooses a new random number k^* to the corresponding parameter a^*1 , he cannot compute the correct parameters a^*3 and a^*4 . Therefore, it is easy to find that the adversary cannot pretend to be a patient.

Similarly, we can get that the adversary A has no ability to pretend to be a doctor because he does not know the RC's secret value u . Therefore, the proposed authentication protocol can resist the impersonation attack.

5.2.4. Stolen Smart Card Attack. In the proposed protocol, every doctor has a smart card to login to the wireless body area networks. Suppose an adversary A picks up or steals a doctor's smart card and extracts the stored secret parameters $\{B_i, V_i, PK_{Di}, L_{Di}, r_i\}$, where $B_i = h(A_i)$, $V_i = A_i \oplus h(DID_i || h(r_i \oplus DPW_i))$, $PK_{Di} = s_{Di} \cdot u \cdot P$, and $L_{Di} = s_{Di} \cdot v \cdot P$. Furthermore, assume that the adversary A eavesdrops the authentication message $\{b_1, b_5, c_2, \text{Cert}_i, T_i\}$ sent by the doctor. Using these obtained parameters, if A wants to pretend to be a doctor and launch an attack, he must try to guess the doctor's password DPW_i to generate the doctor's private key $s_{Di} = h(r_i \oplus DPW_i)$. Without knowing the doctor's password, the adversary A cannot compute the doctor's private key. Then he cannot further generate the correct authentication message. Therefore, it is easy to find that the proposed protocol is resistant to stolen smart card attack.

TABLE 2: The notations and rules of the BAN logic.

Notations	Description
P, Q	A principal
$P \triangleleft X$	P sees X
$P \sim X$	P said X , X was send by P
$P \Rightarrow X$	P has jurisdiction over X
$\xrightarrow{k} P$	k is P 's public key
$P \xleftrightarrow{k} Q$	k is only known to P and Q .
$\#(X)$	X is fresh
$\langle X \rangle_k$	Formulae X is combined with the formulae k
$\{X\}_k$	X is encrypted by the key k
$P \equiv X$	P has faith in the truth of X
Rule 1: message meaning rule	$(P \equiv P \xleftrightarrow{k} Q, P \triangleleft \langle X \rangle_k) / (P \equiv Q \sim X)$ or $(P \equiv \xrightarrow{k} Q, P \triangleleft \{X\}_k) / (P \equiv Q \sim X)$
Rule 2: nonce verification rule	$(P \equiv \#(X), P \equiv Q \sim X) / (P \equiv Q \equiv X)$
Rule 3: jurisdiction rule	$(P \equiv Q \Rightarrow X, P \equiv Q \equiv X) / (P \equiv X)$
Rule 4: decomposition rule	$P \equiv Q \equiv (X, Y) / (P \equiv Q \equiv X)$

5.2.5. User Anonymity. User anonymity is a very important security requirement in the WBAN. To protect the privacy of doctors and patients, the proposed protocol has made the following measures. In the patient side, the random value $k \in Z^*p$ and the timestamp T_j are used in each round of the patient to doctor authentication. The patient's master key s_{Pj} and public key PK_{Pj} are encrypted in a_3, a_4 with k and T_j , respectively. Suppose that the adversary A could intercept the message $\{a_1, c_1, T_j\}$, it is an impossible task for to obtain the patient's fixed master key s_{Pj} and public key PK_{Pj} . Similarly, the adversary A cannot use the message transferred from the doctor to the patient to obtain the doctor's fixed parameters. Consequently, the proposed authentication protocol can achieve the anonymity of the patients and the doctors.

5.2.6. Authentication and Data Integrity. In the proposed scheme, the patient's physiological parameter data and the doctor's query request information demand are encrypted by the hash values $H(a_2)$ and $H(b_2)$, respectively. In addition, the values $h(\text{data})$ and $h(\text{demand})$ are the parameters of a_3 and b_3 , respectively. According to the property of hash, if any bits are modified, the verify equations $\hat{e}(a_3, PK_{Di})? = \hat{e}(PK_{RC}, h(\text{data}) \cdot s_{Di} \cdot a_4)$ and $\hat{e}(b_3, PK_{Pj})? = \hat{e}(PK_{RC}, h(\text{demand}) \cdot s_{Pj} \cdot b_4)$ cannot be established. Consequently, the proposed authentication protocol can check the integrity of the messages transmitted between the doctor and the patient.

5.2.7. Unlinkability and Conditional Traceability. For the adversary A , he could intercept the messages $\{a_1, c_1, T_j\}$ and $\{b_1, b_5, c_2, Cert_i, T_i\}$. However, the random numbers k and r are different in each round of the message authentication. Therefore, it is difficult for the adversary A to trace the messages which were transmitted from the doctor or the patient. On the other hand, the RC has the ability to track the doctor's actual identity through the formula in Step DA-5. Therefore, except for the ability of RC to track the identity of doctors, other entities cannot track the identity of doctors or patients.

6. Performance Analysis

In this section, the performance of the proposed scheme is evaluated in terms of computational cost, and communication overhead, and security requirements. We then compare the proposed scheme with the existing research activities in terms of security and functional features.

6.1. Computation Cost. In the proposed scheme, the computational cost is referred to the time which was consumed in the phase of message generation and verification. The multiplicative cyclic groups used in the proposed scheme are built based on a Type-A elliptic curve, which is defined in the pairing-based cryptography (PBC) library [34]. In addition, we use C language under specific IDE and C/CCC MIRACL Library to implement the related cryptographic operations. To evaluate the computational costs of the proposed scheme, some of the related notations are listed in Table 3.

Our implementation uses a PC with Intel Core i7 CPU 2.6 GHz and 8 GB memory to run the proposed authentication protocol. In our simulation, each randomized ID is 1024 bits, and the size of the ECC point is 160 bits. The execution time for each cryptographic operation is derived after 10 times experiments. The average running time of each cryptographic operation is listed in Table 4. It needs to be explained here that we have ignored the running time of the XOR operation because it is negligible.

In our implementation, the costs of the registration and smart card distribution are not considered since it only runs a limited number of times in the initial stage of the proposed protocol. Table 5 shows a comparison for computation cost between the proposed authentication protocol and the related works. From Table 5, it is obvious that the proposed authentication protocol takes only one point multiplication, one pairing, and one hash function to generate the certificate. And the time of verifying the certificate only needs one hash function, two point multiplication, and one pairing operation. Compared with the related research activities, it is easy to find that the proposed protocol needs a very low computational overhead to complete the authentication process.

TABLE 3: Execution time of the related pairing-based operations.

Notations	Execution time for various operations
T_h	One-way hash function $H(\cdot)$ or $h(\cdot)$
T_{pair}	Bilinear pairing computation
T_{add}	Addition operation of points in ECC
T_{exp}	Exponential operation
T_{mul}	Scalar multiplication of elliptic curve
T_{en}	Symmetric encryption algorithm AES (128-bit key)

TABLE 4: Execution time of the related pairing-based operations.

Encryption element	T_h (ms)	T_{pair} (ms)	T_{add} (ms)	T_{exp} (ms)	T_{mul} (ms)	T_{en} (ms)
Running time	<1	3.61	<1	2.74	1.63	<1

TABLE 5: Execution time of the related pairing-based operations.

Schemes	Time of generating the certificate	Time of verifying the certificate
Wu et al.'s scheme	$3T_h + 4T_{\text{mul}} + T_{\text{en}}$	$4T_h + 4T_{\text{mul}} + T_{\text{pair}} + T_{\text{en}}$
Shen et al.'s scheme	$T_h + 3T_{\text{mul}}$	$T_h + 4T_{\text{pair}} + T_{\text{en}} + 2T_{\text{mul}}$
Das et al.'s scheme	$5T_h + 2T_{\text{en}}$	$4T_h + 2T_{\text{en}}$
Liu et al.'s scheme	$2T_{\text{pair}} + 3T_h + T_{\text{mul}}$	$3T_{\text{pair}} + 3T_h + 3T_{\text{mul}}$
Proposed scheme	$T_h + T_{\text{mul}} + T_{\text{pair}}$	$T_h + 2T_{\text{mul}} + T_{\text{pair}}$

6.2. *Communication Overhead.* To analyze the communication overhead of the proposed authentication protocol, the size of the parameters used in the proposed scheme is shown below. The length of the random number, the point of ECC, the identity, the output of a hash function, and the timestamp are 128 bits, 320 bits, 128 bits, 160 bits, and 32 bits, respectively. We assumed that the length of the physical parameters of the patient *data* and the query request information of the doctor *demand* are 500 bits and 300 bits, respectively.

Under these deliberations, in the patient to doctor authentication phase of the proposed protocol, the patient sends the message $M_1 = \{a_1, c_1, T_j\}$ to the doctor. Similarly, in the doctor to patient authentication phase, the doctor sends the message $M_2 = \{b_1, b_5, c_2, \text{Cert}_i, T_i\}$ to the doctor. These two messages need $320 + 500 + 320 + 320 + 32 + 32 = 1524$ bits and $320 + 320 + 300 + 320 + 32 = 1292$ bits, respectively. In Table 6, we summarize the brief comparison of communication overhead between the proposed scheme and other existing schemes.

Compared with other existing schemes, the proposed scheme's communication cost is similar to that of other related research works. However, the messages in the proposed protocol contain the patient's physical parameter data and the doctor's query request information demand. In other words, the proposed scheme can not only achieve the identity authentication, but also complete the transfer of the patient's physiological data and the data requested by the doctor. Therefore, the proposed protocol is not only efficient in terms of communication overhead in the WBAN system but also has more extra features.

TABLE 6: The comparison of communication cost in different schemes.

Scheme	Number of messages	Communication cost (bits)
Wu et al.'s scheme	3	2112
Shen et al.'s scheme	4	3040
Das et al.'s scheme	2	1536
Liu et al.'s scheme	4	3840
Proposed scheme	2	2816

TABLE 7: The comparison of security requirements.

Scheme	I_1	I_2	I_3	I_4	I_5	I_6	I_7
Wu et al.'s scheme	√	√	√	√	√	×	√
Shen et al.'s scheme	√	×	√	√	×	√	√
Das et al.'s scheme	√	√	√	×	√	√	×
Liu et al.'s scheme	√	√	√	√	×	√	√
Proposed scheme	√	√	√	√	√	√	√

Note. I_1 : replay attack; I_2 : impersonation attack; I_3 : privileged insider attack; I_4 : secure mutual authentication; I_5 : message integrity and confidentiality; I_6 : user privacy; I_7 : loss of device attack.

6.3. *Security Requirements.* We compare the proposed authentication protocol with the related authentication schemes in terms of security requirements such as replay attack, impersonation attack, secure mutual authentication, message integrity, and confidentiality. The detailed comparison of various security attacks and functions is shown in Table 7. The comments from Table 7 show that our

authentication protocol not only gives the support of much more functionality but also overcomes more security weaknesses.

7. Conclusion

In this article, an efficient and privacy-preserving authentication protocol for the WBAN is presented. In the proposed authentication scheme, the doctor and the patient are anonymously authenticated by each other before sending the patient-related information (the patient's physical parameters or the doctor's query request). The security analysis showed that the proposed authentication protocol could provide resistance against common attacks such as replay attack, impersonation attack, and eavesdropping attack. The proposed authentication scheme takes very little cost for signature and certificate authentication, which is essential for the WBAN-based applications. Moreover, the proposed scheme gives an effective privacy and tracking method to disclose the actual identification of the malicious doctor to improve the usability of the WBAN. The performance analysis showed that the proposed scheme is efficient in terms of computational cost and communication cost. It is more appropriate for practical WBAN-based applications. The future extension of this article is to provide an authentication method that can transmit a larger amount of data for the patient in an efficient manner.

Data Availability

The data used to support the findings of this study are available at <https://crypto.stanford.edu/pbc/>.

Conflicts of Interest

None of the authors have any conflicts of interest.

Acknowledgments

This research was supported by the National Natural Science Foundation of China (Grant nos. 61772477 and U1804263) and the Key Scientific Research Projects of Colleges and Universities in Henan Province (no. 16A520075).

References

- [1] S. H. Islam, M. Azees, N. Kumar et al., "Efficient and secure anonymous authentication with location privacy for IoT-based WBANs," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 4, pp. 2603–2611, 2020.
- [2] M. Hussain, A. Mehmood, S. Khan et al., "Authentication techniques and methodologies used in wireless body area networks," *Journal of Systems Architecture*, vol. 101, 2019.
- [3] M. Umar, Z. Wu, and X. Liao, "Mutual authentication in body area networks using signal propagation characteristics," *IEEE Access*, vol. 8, pp. 66411–66422, 2020.
- [4] M. Shuai, L. Xiong, C. Wang, and N. Yu, "Lightweight and privacy-preserving authentication scheme with the resilience of desynchronisation attacks for WBANs," *IET Information Security*, vol. 14, no. 4, pp. 380–390, 2020.
- [5] X. Liu, R. Zhang, and M. Zhao, "A robust authentication scheme with dynamic password for wireless body area networks," *Computer Networks*, vol. 161, pp. 220–234, 2019.
- [6] V. Odelu, S. Saha, R. Prasath, L. Sadineni, M. Conti, and M. Jo, "Efficient privacy preserving device authentication in WBANs for industrial e-health applications," *Computers & Security*, vol. 83, pp. 300–312, 2019.
- [7] K.-A. Shim, "Universal forgery attacks on remote authentication schemes for wireless body area networks based on Internet of Things," *IEEE Internet of Things Journal*, vol. 6, no. 5, pp. 9211–9212, 2019.
- [8] D. He and D. Wang, "Robust biometrics-based authentication scheme for multiserver environment," *IEEE Systems Journal*, vol. 9, no. 3, pp. 816–823, 2015.
- [9] S. Chatterjee, A. K. Das, and J. K. Sing, "A novel and efficient user access control scheme for wireless body area sensor networks," *Journal of King Saud University-Computer and Information Sciences*, vol. 26, no. 2, pp. 181–201, 2014.
- [10] J. Liu, Z. Zhang, X. Chen, and K. S. Kwak, "Certificateless remote anonymous authentication schemes for WirelessBody area networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 2, pp. 332–342, 2014.
- [11] T.-F. Lee and M. Chen, "Lightweight identity-based group key agreements using extended chaotic maps for wireless sensor networks," *IEEE Sensors Journal*, vol. 19, no. 22, pp. 10910–10916, 2019.
- [12] A. K. Das, S. Chatterjee, and J. K. Sing, "A new biometric-based remote user authentication scheme in hierarchical wireless body area sensor networks," *Ad Hoc and Sensor Wireless Networks*, vol. 28, no. 3-4, pp. 221–256, 2015.
- [13] C. Wang and Y. Zhang, "New authentication scheme for wireless body area networks using the bilinear pairing," *Journal of Medical Systems*, vol. 39, no. 11, p. 136, 2015.
- [14] H. Debiao, S. Zeadally, N. Kumar, and J.-H. Lee, "Anonymous authentication for wireless body area networks with provable security," *IEEE Systems Journal*, vol. 11, no. 4, pp. 2590–2601, 2017.
- [15] J. Liu, L. Zhang, and R. Sun, "1-RAAP: an efficient 1-round anonymous authentication protocol for wireless body area networks," *Sensors*, vol. 16, no. 5, p. 728, 2016.
- [16] X. Li, J. Peng, F. Wu, M. Karuppiyah, and K.-K. Raymond Choo, "An enhanced 1-round authentication protocol for wireless body area networks with user anonymity," *Computers & Electrical Engineering*, vol. 61, pp. 238–249, 2017.
- [17] J. Shen, S. Chang, J. Shen, Q. Liu, and X. Sun, "A lightweight multi-layer authentication protocol for wireless body area networks," *Future Generation Computer Systems*, vol. 78, no. 3, pp. 956–963, 2018.
- [18] X. Liu, C. Jin, and F. Li, "An improved two-layer authentication scheme for wireless body area networks," *Journal of Medical Systems*, vol. 42, no. 8, pp. 143–154, 2018.
- [19] M. Wazid, A. K. Das, and A. V. Vasilakos, "Authenticated key management protocol for cloud-assisted body area sensor networks," *Journal of Network and Computer Applications*, vol. 123, pp. 112–126, 2018.
- [20] S. Qiu, G. Xu, H. Ahmad, and L. Wang, "A robust mutual authentication scheme based on elliptic curve cryptography for telecare medical information systems," *IEEE Access*, vol. 6, pp. 7452–7463, 2017.
- [21] J. Shen, Z. Gui, S. Ji, J. Shen, H. Tan, and Y. Tang, "Cloud-aided lightweight certificateless authentication protocol with anonymity for wireless body area networks," *Journal of Network and Computer Applications*, vol. 106, no. 6, pp. 117–123, 2018.

- [22] M. Shuai, B. Liu, N. Yu, L. Xiong, and C. Wang, "Efficient and privacy-preserving authentication scheme for wireless body area networks," *Journal of Information Security and Applications*, vol. 52, Article ID 102499, 2020.
- [23] M. Fotouhi, M. Bayat, A. Das, H. Far, S. Pournaghi, and M. A. Doostari, "A lightweight and secure two-factor authentication scheme for wireless body area networks in health-care IoT," *Computer Networks*, vol. 177, Article ID 107333, 2020.
- [24] D. Kumar, H. S. Grover, and Adarsh, "A secure authentication protocol for wearable devices environment using ECC," *Journal of Information Security and Applications*, vol. 47, pp. 8–15, 2019.
- [25] S. Jegadeesan, M. Azees, N. Ramesh Babu, U. Subramaniam, and J. D. Almahles, "EPAW: efficient privacy preserving anonymous mutual authentication scheme for wireless body area networks (WBANs)," *IEEE Access*, vol. 8, pp. 48576–48586, 2020.
- [26] D. Dolev and A. Yao, "On the security of public key protocols," *IEEE Transactions on Information Theory*, vol. 29, no. 2, pp. 198–208, 1983.
- [27] B. Narwal and A. K. Mohapatra, "A survey on security and authentication in wireless body area networks," *Journal of Systems Architecture*, vol. 113, Article ID 101883, 2020.
- [28] J. Cui, X. Zhang, H. Zhong, J. Zhang, and L. Liu, "Extensible conditional privacy protection authentication scheme for secure vehicular networks in a multi-cloud environment," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 1654–1667, 2020.
- [29] W. Tan, J. Zhang, Y. Zhang et al., "A PUF-based and cloud-assisted lightweight Authentication for multi-hop body area network," *Tsinghua Science and Technology*, vol. 26, no. 1, pp. 36–47, 2021.
- [30] X. Li, M. H. Ibrahim, S. Kumari, A. K. Sangaiyah, V. Gupta, and K.-K. R. Choo, "Anonymous mutual authentication and key agreement scheme for wearable sensors in wireless body area networks," *Computer Networks*, vol. 129, pp. 429–443, 2017.
- [31] J. Zhang, L. He, Q. Zhang et al., "Pseudonym-based privacy protection scheme for participatory sensing with incentives," *Ksii Transactions on Internet & Information Systems*, vl.vol. 10, no. 11, pp. 5654–5673, 2016.
- [32] S. F. Aghili, H. Mala, P. Kaliyar, and M. Conti, "SecLAP: secure and lightweight RFID authentication protocol for medical IoT," *Future Generation Computer Systems*, vol. 101, pp. 621–634, 2019.
- [33] Z. Ali, A. Ghani, I. Khan, S. A. Chaudhry, S. K. H. Islam, and D. Giri, "A robust authentication and access control protocol for securing wireless healthcare sensor networks," *Journal of Information Security and Applications*, vol. 52, pp. 1–14, Article ID 102502, 2020.
- [34] B. Lynn, "Pbc library—the pairing-based cryptography library," 2007, <https://crypto.stanford.edu/xbc/>.

Research Article

Blockchain-Based Key Management and Green Routing Scheme for Vehicular Named Data Networking

Hao Liu,¹ Rongbo Zhu ,² Jun Wang,¹ and Wengang Xu¹

¹College of Computer Science, South-Central University for Nationalities, Wuhan 430074, China

²College of Informatics, Huazhong Agricultural University, Wuhan 430070, China

Correspondence should be addressed to Rongbo Zhu; rongbozhu@163.com

Received 7 April 2021; Revised 13 June 2021; Accepted 30 June 2021; Published 8 July 2021

Academic Editor: Jie Cui

Copyright © 2021 Hao Liu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Due to the distributed and dynamic characteristics of the Internet of Vehicles (IoV) and the continuous growth in the number of devices, content-centric decentralized vehicular named data networking (VNDN) has become more suitable for content-oriented applications in IoV. However, the existing centralized architecture is prone to the failure of single points, which results in trust problems in key verification between cross-domain nodes and consuming more power and reducing the lifetime. Focusing on secure key management and power-efficient routing, this article proposes a blockchain-based key management and green routing scheme for VNDN. A blockchain-based key management scheme is presented to achieve secure and efficient distribution and verification of keys. Specifically, all trusted agencies (TAs) form a consortium blockchain for storing public key hashes to ensure the authenticity of users' public keys. A green global routing scheme based on node relaying pressure (GGNRP) is proposed to save power consumption and reduce the forwarding delay. A new node relay pressure metric is introduced to assist with routing decisions. Detailed experiments and analysis show that, compared with the existing scheme, the proposed scheme can achieve secure key management and GGNRP can decrease the power consumption and average delay by 15.8% and 63.2%, respectively.

1. Introduction

Internet of Vehicles (IoV) is the backbone network of future intelligent transportation systems, and it promises to improve overall traffic efficiency and road safety by enabling the interaction of recreational and safety information through vehicle-to-everything (V2X) communication [1]. To date, vehicles in IoV have relied on IP addresses to find terminals and establish end-to-end communication, regardless of the type of application [2]. Due to the distributed operation, limited bandwidth, and high-speed mobility of nodes and the dynamic network topology of IoV, it is difficult for IoV network links to maintain robustness, durability, and stability [3, 4]. In addition, as the number of Internet of Things (IoT) devices continues to grow, IP addresses are becoming increasingly scarce, making it very difficult to assign IP addresses to IoV devices with high mobility [5]. As a result, a large gap is created between host-based TCP/IP architectures and content-centric IoV applications. Since most of the communication between vehicles focuses on the content

rather than the content carrier, the combination of IoV and named data networking (NDN) becomes possible, resulting in vehicular named data networking (VNDN) [6].

NDN is an important candidate for next-generation Internet architecture, where everything, including hosts and data, is named according to hierarchical naming rules. These names replace the role of IP addresses and data transmission switches from a host-to-host approach to data-oriented communication [7]. In addition, NDN caches content in network routers, which allows content requests to be satisfied at the edge of the network, thereby greatly reducing the delay of content delivery [8]. Therefore, NDN is very suitable for providing a reliable transmission solution for IoV communications with high mobility and intermittent connections. However, because of the data-centric feature of NDN, secure communication in NDN has new security requirements [9]. In the content-centric IoV, vehicles may request traffic information (e.g., traffic accident information and road information) for efficient data sharing to optimize road utilization. However, malicious nodes in VNDN may

spread false information to cause traffic congestion or accidents [10]. Therefore, consumers in IoV should care not only about the sender of the data request but also about the producer of the data packet. The packet must be published by an authenticated producer and be unable to be modified by other producers.

To verify the producer's identity information and data integrity, the producer should sign the content so that the name can be effectively and safely bound to the data. In this way, consumers and routers can verify the signature and determine the source of the data, which allows consumers to trust the received data packets. Most existing NDNs use a hierarchical key trust model [11], in which the root key is used as a well-known trust anchor to provide a digital signature on the domain secret key. The key of each domain digitally signs the public key of the user in the domain, and then the user key signs the public key of its device and application. To verify the authenticity of the public key, one can retrieve the secret key chain using the key name. In principle, this method avoids the generation of false messages, but in the application of IoV, there are still some challenges [12]: (1) as a centralized service centre, the root key may be subject to attacks and tampering, which can lead to a single point of failure. Especially in the case of cross-domain key verification, since each domain is relatively independent, it is difficult for each domain to verify the authenticity of the keys issued by the other domains without a trust anchor; (2) since verification needs to traverse the secret key chain, the process of retrieval and verification requires considerable additional overhead, which cannot meet the low-delay requirements of IoV.

On the other hand, the successful implementation of IoV requires a large number of wireless sensors to form a wireless sensor network (WSN) for efficient and fast information transfer. However, the sensors have limited energy and cannot be recharged once they are deployed [13, 14]. The higher the energy efficiency is, the longer the running time of WSN is. The research results show that communication consumes the most energy among many factors that consume energy in WSN [15]. While routing determines the forwarding path between the sender and receiver, effective routing minimizes the communication cost and maximizes the survival time of the wireless sensor network.

In recent years, blockchain technology has been widely used in different fields, such as public key infrastructure (PKI), domain name server (DNS), and IoV [16]. Blockchain ensures that data can be tracked and cannot be easily tampered with through distributed data storage and consensus mechanisms, which guarantees the integrity and authenticity of the participating nodes. To improve the information transmission of blockchain nodes, the combination scheme of blockchain and VNDN was proposed [17]. Therefore, this article proposes a blockchain-based key management and green routing scheme for VNDN, which aims to achieve safe and reliable VNDN key authentication and management while maximizing the use time of wireless sensors. First, a blockchain-based key management scheme is designed to set the management node of each domain as a blockchain node and use blockchain to manage the public

keys of different domains to avoid network paralysis due to the failure of a single point. Second, to prevent the premature death of nodes close to the base station (BS) and prolong the survival time of sensors, the concept of node relay pressure is proposed, and a green global routing scheme based on node relaying pressure (GGNRP) is designed. In GGNRP, the source node obtains a green global route for data transmission based on the node-to-BS path information and node energy information stored in the BS, which avoids the routing hole problem that is widely found in planar routing.

The contributions of this article are summarized as follows:

- (1) In this article, a blockchain-based key management scheme is proposed to solve the mutual trust problem between different domain nodes. The scheme reduces the number of signature verifications and shortens the time delay of key acquisition and verification, making the NDN more suitable for IoV.
- (2) To reduce the transmission delay of VNDN, a green global routing scheme based on node relaying pressure is designed. This scheme uses the path transmission delay and the node relaying pressure value as metrics for routing decisions, which ensures low delay while protecting the nodes with high communication load and low residual energy in VNDN.

The rest of this article is organized as follows. Related work is presented in Section 2. Section 3 details the proposed blockchain-based key management scheme. GGNRP is presented in Section 4. The experimental results are presented in Section 5, followed by the conclusions in Section 6.

2. Related Work

2.1. Security in NDN. NDN is expected to change the architecture of the Internet. For this reason, researchers hope to introduce NDN into IoV to enhance the scalability, reliability, and security of IoV [18]. However, the security requirements of IoV are still difficult to meet due to the high dynamic topology, high mobility, delay, and propagation content [19]. On the other hand, NDN still has various security and privacy issues [20], such as naming, signature, and cache privacy, which makes the establishment of VNDN challenging. Several works have designed solutions to address security issues from the perspective of NDNs [21–23]. Song et al. [21] proposed a smart contract-based trusted content retrieval mechanism for NDNs. This mechanism uses smart contract-based content and a repository of information trusted by producers and provides content retrieval and name resolution services for content consumers. A blockchain-based effective identifier management scheme in the NDN environment was proposed in [22]. This scheme uses the content name of an identifier to create transactions to protect the identifier of a specific user and realizes secure storage and management through this identifier segmentation management technology. A blockchain-based

hierarchical identity-based security mechanism was proposed for NDN to maintain data-oriented authentication [23]. However, most of the existing solutions do not take into account the characteristics of IoV, making them inapplicable in high mobility and low-delay IoV.

For IoV, most existing schemes focus on routing and relaying [24, 25]. To maximize the possibility for users to retrieve the desired content, Mauri et al. [24] formulated this problem as an integer linear programming (ILP) problem and showed how to optimally distribute content in IoV while considering the available storage capacity and available link capacity. In [25], an active data distribution scheme was proposed to push key content to one-hop neighbors in VNDN.

Focusing on information security and privacy preservation in vehicular ad hoc networks, a full session key agreement scheme was proposed based on chaos mapping [26]. To achieve fast authentication during the message verification process, a novel Chinese remainder theorem (CRT)-based conditional privacy-preserving authentication scheme was presented [27]. To manage keys efficiently, a scalable solution was proposed for key and trust management of devices [28], with the combination of blockchain and software-defined networking (SDN) that is able to store the public keys of devices on the blockchain and route the network traffic efficiently. To address the low security and communication efficiency in the blockchain, a key secret-sharing scheme was proposed based on generative adversarial networks (GANs), which view the secret as an image during the secret-sharing process [29]. However, since the security aspects of VNDNs have not been extensively studied, communication in IoV can be subject to many security threats, such as denial of service (DoS) attacks, worm attacks, disinformation attacks, replay attacks, timing attacks, single points of failure, and content poisoning attacks.

2.2. Green Routing Protocol. According to the network structure, the existing green routing protocols can be divided into two types. One type is hierarchical routing, such as the low-energy adaptive clustering hierarchy (LEACH) [30], power-efficient gathering in sensor information systems (PEGASIS) [31], and the energy-efficient concentric clustering routing scheme (EECCRS) [32]. In these routing protocols, the network is clustered into groups according to the distribution of the system, there are several nodes in each cluster, and each node belongs to only one cluster. There is a cluster head (CH) in each cluster, and the CH needs to collect and process the data from the cluster members that are in the same cluster. The processed data are transmitted to the base node directly or indirectly. Hierarchical routing protocols have excellent expansibility and are easy to manage. However, forwarding data will cost tremendous energy. The other type of green routing protocol is flat routing, such as node spatial distribution (NSD) [33], geographic routing oriented sleep scheduling (GSS) [34], energy-balanced routing protocol (EBRP) [35], energy savings via

opportunistic routing (ENS_OR) [36], and the energy-balanced routing method based on forward-aware factor (FAF-EBRM) [37]. Unlike hierarchical routing protocols, all nodes in flat routing protocols are the same, and each node communicates with the base node in a multihop manner. However, there are issues of poor extensibility and hole problems in flat routing.

The schemes mentioned above provide effective solutions for VNDN, but their applicability in VNDN with high mobility and high data volume is limited. Therefore, this article tries to fill this gap and proposes a blockchain-based key management and green routing scheme for VNDN, in which keys can achieve safe and efficient management and authentication by using blockchain. Additionally, it aims to decrease power consumption and delay.

3. Blockchain-Based Key Management

3.1. System Model. In this section, a blockchain-based key management scheme is introduced to solve the problem of lack of trust in interdomain nodes and to improve the authentication efficiency of key management. The system model of blockchain-based key management is shown in Figure 1, which contains the main parts described as follows.

3.1.1. Trusted Agency (TA). A trusted third-party authority that provides services for the domain is mainly used to generate public/private key pairs (PB_{Uk} , PV_{Uk}) for user i in the domain and public/private key pairs (PB_{Dk} , PV_{Dk}) for the domain. Meanwhile, the TA joins the consortium blockchain as a node of the blockchain to manage the generated keys securely and efficiently. Each TA is responsible for managing one domain.

3.1.2. Routing Node. VNDN routing nodes have relaying, caching, and broadcasting functions. The main function is to relay interest packets to nodes that have data and trace data packets back to consumers.

3.1.3. Domain. In an institution or organization, each domain contains multiple VNDN users and uses its private key to sign the users in the domain for authentication and to ensure the trustworthiness of the users. Each domain has a domain name that serves as a unique identifier in VNDN. The name of the domain can be expressed as follows:

$$|Public\ key|Hierarchy|PublicKeyHash|Version, \quad (1)$$

where *Public key* denotes the public key name of the domain, *Hierarchy* denotes the domain hierarchy to which the name belongs, *PublicKeyHash* is the hash of the domain public key, and *Version* denotes the version number.

3.1.4. User. A data requester or data producer consists of intelligent vehicles and terminal equipment in VNDN. The name of the user can be expressed as follows:

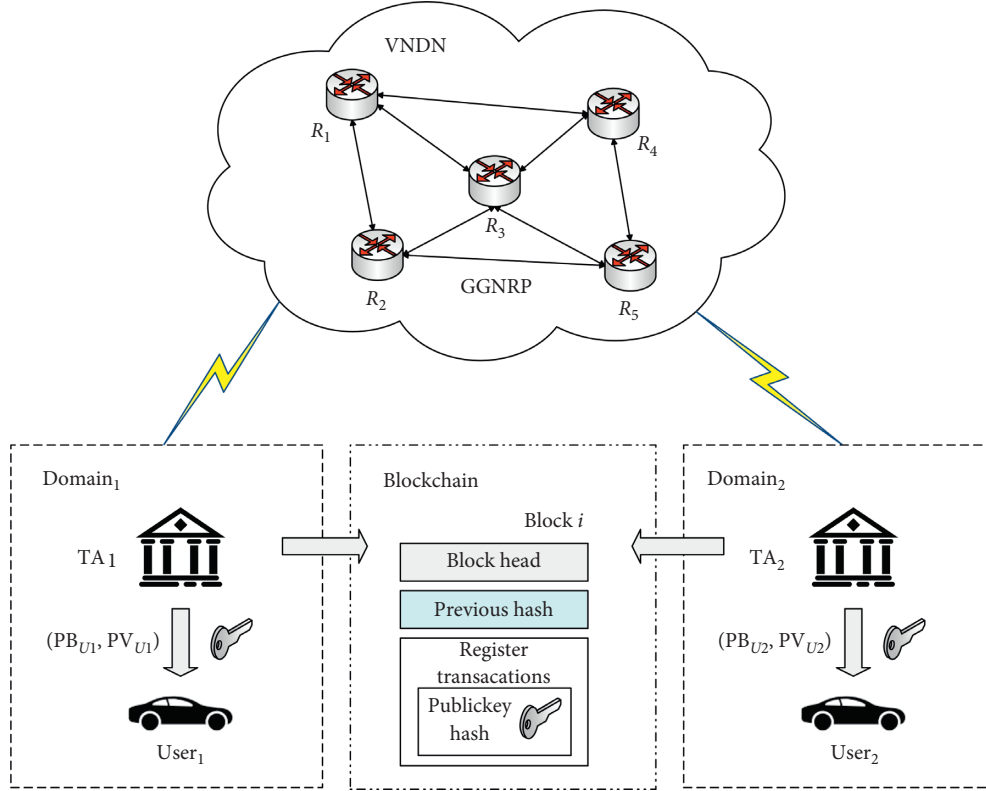


FIGURE 1: The system model.

$$|Global\ route|Hierarchy|PublicKeyHash|BlockLocation|TransactionHash|Version, \quad (2)$$

where *Global route* indicates the global and routable name for guiding the routing policy, *Hierarchy* denotes the user hierarchy to which the name belongs, *PublicKeyHash* indicates the hash of the user's public key, which will also be stored in the blockchain to verify the authenticity of the corresponding public key, *BlockLocation* indicates the location of the public key hash in the blockchain for fast retrieval, *TransactionHash* indicates the hash of the user's registered transaction, and *Version* indicates the version number.

Different from the existing schemes, the blockchain includes the block head and register transactions, the unique license, and the transaction record. As shown in Figure 1, each block contains a block header and a block body. The block header contains the hash value of the previous block, the timestamp, the hash value of the current block, and the root hash. The block body contains details of the transaction.

The proposed blockchain-based key management scheme has the following features.

- (1) Integrity: all data are required to be signed by the data producer, and the data requester can easily verify the signature to be sure that the data have not been modified during the relaying process.
- (2) Confidentiality: the content of any transaction message should be protected by asymmetric

cryptography and digital signatures and should not be affected by any other entities.

- (3) Reliability: after confirming that the data have not been modified, consumers can determine the source of the data so that they can trust the acquired data.
- (4) Authentication: authentication services are the basis for achieving trustworthiness. After verifying that the received data have not been modified, the blockchain network and TA are used to verify the legitimacy of the data producer.
- (5) Efficiency: the key retrieval and verification process is reduced to provide efficient key management and certification while providing basic services.

3.2. Key Management and Authentication. The blockchain-based key management design focuses on two main aspects: one is to verify the integrity of the data and the other is to quickly verify the credibility of the data packet. The producer uses the private key to sign the data packet and send it to the consumer. After the consumer receives the data packet, it first uses the producer's public key to verify the signature to ensure the integrity of the data. Then, it needs to authenticate the producer. If the data packet comes from a legitimate producer, the consumer trusts the data packet.

The designed blockchain-based key management scheme is divided into four main parts: system initialization, blockchain creation, packet transmission, and producer authentication.

Step 1: system initialization: system initialization is performed using an elliptic curve digital signature algorithm and asymmetric cryptography to ensure data confidentiality and integrity. TA_i first issues the public/private key pairs (PB_{Dk}, PV_{Dk}) , (PB_{Uk}, PV_{Uk}) for the domain and user i within the domain.

Step 2: blockchain creation: user i creates a registration transaction and writes its public key hash to the transaction. The registered transaction is then sent to TA_i to verify its legitimacy and is added to the blockchain. TA_i verifies the legitimacy of the transaction, signs it, and broadcasts it to other blockchain nodes for consensus. The consensus nodes use the practical Byzantine fault tolerance (PBFT) consensus algorithm to conduct the consensus process on the transaction. After passing the consensus process, the registered transaction is uploaded to the consortium blockchain, and the public key hash of user i is stored in it. After that, TA_i returns the $BlockLocation_i$ and $TransactionHash_i$ to user i , who writes them into the name.

Step 3: packet transmission: the consumer sends an interest packet to the router to request the content it needs. If the data are cached in the local storage of the intermediate router, the router returns a data packet to the consumer. Otherwise, the router forwards the Interest packet to the producer. Finally, the data packet is sent back to the consumer by the producer in the same way. In the asymmetric cryptographic scheme, the decryption $Ver_{PB_k}(\cdot)$ of the digitally signed data using the public key of sender k is as follows:

$$Ver_{PB_k}(\text{Sig}_{PV_k}(H(m))) = H(m), \quad (3)$$

where $\text{Sig}_{PV_k}(\cdot)$ is the digital signature using the private key of sender k and $H(m)$ is the hash digest of message m .

Step 4: producer authentication: after the consumer receives the data packet, it uses the producer's public key to decrypt and verify the digital signature. However, only the public key from a legitimate producer can be trusted by the consumer. Therefore, the authenticity of the public key must be verified first. The consumer first checks the $BlockLocation_i$ and $TransactionHash_i$ fields in the name to quickly find the location of the registered transaction containing the hash of the public key. Then, the consumer obtains the public key hash stored in the blockchain from the "registration transaction" and calculates the obtained user's public key hash with the SHA-256 algorithm. After that, the two are compared and if the hash value is the same, the public key is proven to be true. Otherwise, the obtained public key is not the public key issued by a legitimate user.

4. Green Global Routing Scheme Based on the Node Relaying Pressure

4.1. Basic Definition. The symbols used in this paper are shown in Table 1.

Definition 1. (the maximal minimum hop) Assume that V is the set of nodes in VNDN and each node v_i 's minimum hop is known as hop_i ; then, the maximal minimum hop m is

$$m = \max(\text{hop}_i), \quad v_i \in V. \quad (4)$$

Definition 2. (the node relaying pressure) If v_i 's minimum hop is hop_i , m is the maximal minimum hop of the network and E_i is the residual energy of v_i , then v_i 's relaying pressure press_i is

$$\text{press}_i = \frac{2(m - \text{hop}_i) + 1}{E_i}. \quad (5)$$

Definition 3. (the set of candidate relaying node) The set of neighbor of v_i is denoted as nbor_i , and v_i 's minimum hop is hop_i ; then, the set of v_i 's candidate nodes cand_i is

$$\text{cand}_i = \sum_{v_j \in \text{nbor}_i} \text{nbor}_i(\text{hop}_j < \text{hop}_i). \quad (6)$$

Considering that there are 18 nodes and one BS in VNDN shown in Figure 2, each node's minimum hop is known. In this network, hop_1 , hop_2 , hop_4 , hop_{13} , and hop_{18} are all 3. Hop_3 , hop_5 , hop_8 , hop_9 , hop_{12} , hop_{14} , hop_{15} , and hop_{16} are 2. Hop_6 , hop_7 , hop_{10} , hop_{11} , hop_{17} , and hop_{19} are 1. According to the definitions, the maximal minimum hop m is 3. For node v_8 , $\text{hop}_8 = 2$, $m = 3$, we can obtain $\text{press}_8 = 3/E_8$, and $\text{nbor}_8 = \{v_4, v_7, v_{12}, v_{16}, v_{17}\}$, where hop_7 , hop_{17} are less than 2; hence, we have $\text{cand}_8 = \{v_7, v_{18}\}$.

An example of communications between nodes and a BS is shown in Figure 3, where the red dotted circle is the communication range of the nodes. To facilitate the analysis, we assume that all nodes in this model have the same maximal transmission radius r . The network is composed of n nodes and one BS. When their distance is larger than r , they are unable to send packets to each other directly. The tasks of the nodes are to collect data in their deployed areas and transmit the collected data to the BS. When a node has packets to send, it communicates with the BS in a multihop way. As shown in Figure 3, when v_1 has data to forward, the packet can reach the BS with the help of v_5 and v_6 . In the initial phase, all nodes have the same energy, and the energy cost is related to the number of packets, the size of each packet, and the forwarding distance.

For the proposed GGNNRP, the energy cost E_R of receiving k bits is

$$E_R = E_{\text{elec}} * k, \quad (7)$$

where k is the size of the data packet and E_{elec} is the energy consumption that a node uses to send or receive one-bit data.

TABLE 1: Notations.

Symbol	Description
V	The set of sensor nodes in WSNs
r	The communication radius
m	The maximal minimum hop in network
E_T	The energy cost of transmission node
E_R	The energy cost of the receiving node
E_{elec}	The energy cost per bit
E_{amp}	The energy cost of signal amplification
hop_i	The minimum hop of v_i
$nbor_i$	The set of neighbors of v_i
$cand_i$	The set of candidate relaying nodes
p_{ij}	The path between node v_i and node v_j
$cost_{ij}$	The energy cost of p_{ij}
$press_i$	The relaying pressure of v_i
max_{ij}	The maximal relaying pressure in p_{ij}

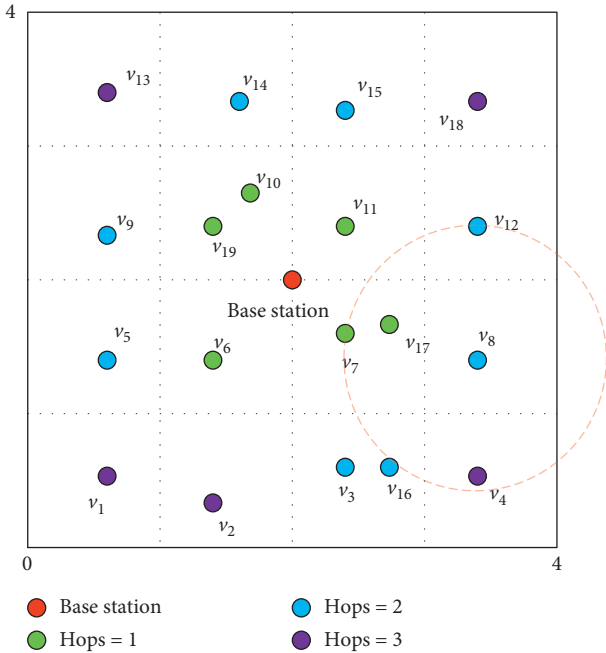


FIGURE 2: The minimum hop of each node.

The transmitting node also needs to consider the propagation loss, and the energy cost E_T of sending k bits to the receiving node is

$$E_T = E_{elec} * k + E_{amp} * k * d^\beta, \quad (8)$$

where E_{elec} represents the signal amplification cost, β is the wave loss factor, and d is the distance from the transmitting node to the receiving node.

4.2. GGNRP Process. The proposed GGNRP consists of two main phases: the routing establishment phase and the data forwarding phase. The flowchart of routing establishment is shown in Figure 4.

Each node obtains its neighbors by broadcasting, and then GGNRP calculates every node's minimum hop by broadcasting several times. Every node computes its

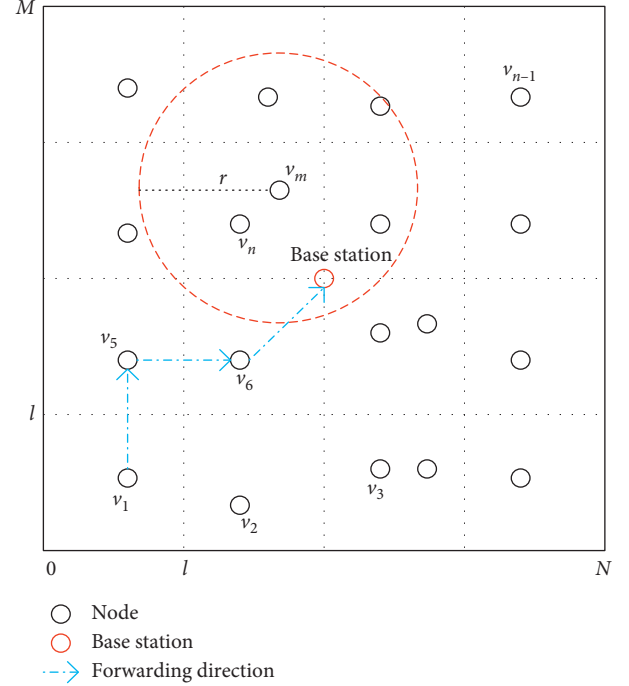


FIGURE 3: An example of communications between nodes and a BS.

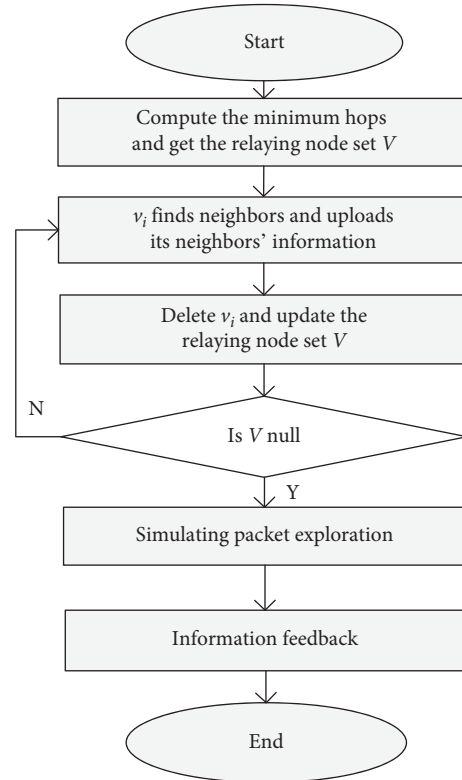


FIGURE 4: The flowchart of routing establishment.

candidate relaying nodes according to its neighbors' minimum hops and its minimum hops. After that, each node uploads its candidate relaying node set, the energy cost of the communication between itself and its candidate set, and its ID to the BS. Then, the BS simulates the process of nodes

sending exploring packets to the BS. The exploring packet that records its forwarding trace is transmitted only to the candidate relaying nodes of the nodes where the packet is. When the BS has computed all the nodes' shortest paths to itself, it will feed all path IDs and relevant information back to the related nodes. GGNRP avoids routing holes by limiting the forwarding objects in exploring data packets.

The routing establishment phase includes the following parts:

Step 1: calculating the minimum hop: GGNRP computes every node's minimum hop by broadcasting several times. Before this step, the minimum hops of the nodes in the network are all an unreachable number x . First, the BS broadcast packet has its maximum communication range, and the packet contains a number that is used to help the nodes compute their minimum hop, which is 1. If the number in the packet is less than its minimum hop, the node will change its minimum hop to this number. Then, the nodes whose minimum hops are equal to the number in the packet will broadcast new packets to their neighbors, and the number in the new packet is one larger than the old number. This process is repeated until there are no nodes whose minimum values are x .

As shown in Figure 5, $v_6, v_7, v_{10}, v_{11}, v_{17}$, and v_{19} are in the communication range of the BS. After BS broadcasting the messages, $v_6, v_7, v_{10}, v_{11}, v_{17}$, and v_{19} have the same minimum hop of 1. Then, $v_6, v_7, v_{10}, v_{11}, v_{17}$, and v_{19} begin to broadcast data. Similarly, $v_3, v_5, v_8, v_9, v_{12}, v_{14}, v_{15}$, and v_{16} set their minimum hop to 2. v_1, v_2, v_4, v_{13} , and v_{18} will achieve the same minimum hop of 3. According to Definition 3, we can obtain the candidate relaying nodes of each node in Figure 5 as shown in Table 2.

Step 2: simulating packet exploration: after Step 1, the BS has already collected enough information to finish the rest of the work in the routing establishment phase. GGNRP finds all the shortest paths to avoid additional energy consumption.

The format of the exploring packet is shown in Table 3, which contains the multicast objects and the forwarding trace. In the exploration process, the exploring packet always takes the candidate relaying nodes of the nodes that the packet is in as the multicast objects. When the exploring packet reaches a node, it will record its ID in its forwarding trace and update its multicast objects as the node. When the BS appears in the multicast objects of the packet, the packet will be transmitted to the BS directly.

The process of v_4 's packet exploration is shown in Figure 6, where the subscript of the packet indicates the trace of the packet. Nodes v_3, v_8 and v_{16} are the candidate relaying nodes of v_4 , so the $packet_4$ updates its content to $\{\{v_3, v_8, v_{16}\}, \{v_4\}\}$, and v_4 forwards it to v_3, v_8 and v_{16} . As $packet_{4-3}$ reaches v_3 , it updates its content to $\{\{v_7\}, \{v_4, v_3\}\}$, and v_3 forwards it to v_7 . The BS is the candidate relaying node of v_7 , and $packet_{4-3-7}$

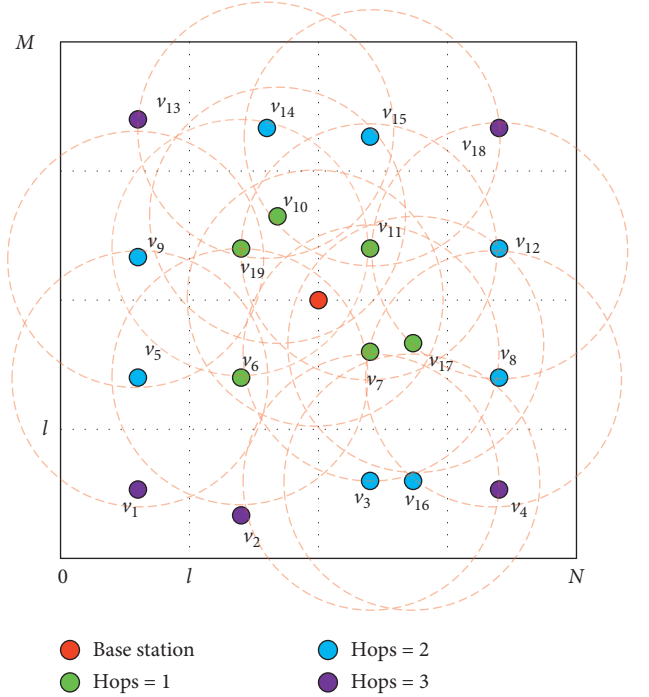


FIGURE 5: Compute the minimum hop.

TABLE 2: The candidate relaying nodes.

Node	The candidate relaying nodes
v_1	v_5
v_2	v_3
v_3	v_7
v_4	v_3, v_8, v_{16}
v_5	v_6
v_6	v_s
v_7	v_s
v_8	v_7, v_{17}
v_9	v_{19}
v_{10}	v_s
v_{11}	v_s
v_{12}	v_{11}, v_{17}
v_{13}	v_{14}
v_{14}	v_{10}, v_{19}
v_{15}	v_{10}, v_{11}
v_{16}	v_7, v_{17}
v_{17}	v_s
v_{18}	v_{12}, v_{15}
v_{19}	v_s

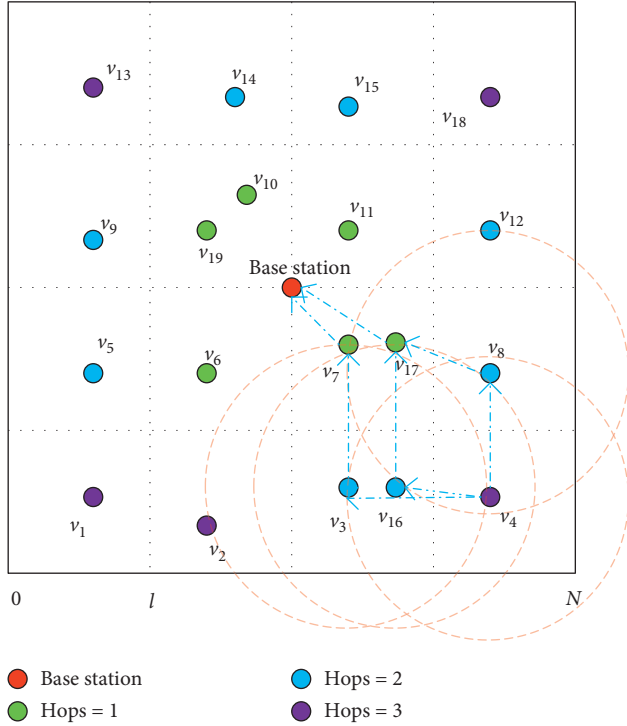
reaches the BS through v_7 . Similarly, $packet_4$ is sent to v_8 , $packet_{4-8}$ is forwarded to v_{17} , and $packet_{4-8-17}$ finally reaches the BS. Then, $packet_4$ is sent to v_{16} , $packet_{4-16}$ is forwarded to v_{17} , and $packet_{4-16-17}$ reaches the BS.

There are three shortest paths between v_4 and the BS, which are shown in Table 4.

Step 3: information feedback: the BS numbers all the paths that have only IDs. Then, the BS feeds the path IDs back to the related nodes. For example, as the BS obtains all the paths of v_4 , it will feed information back

TABLE 3: The format of the exploring packet.

Multicast objects	Forwarding trace
cand _i	$\{v_0, \dots, v_i\}$

FIGURE 6: The process of v_4 's packet exploration.TABLE 4: The shortest path from v_4 to the BS.

Path ID	Path trace
P_{40}	$v_4 \rightarrow v_3 \rightarrow v_7 \rightarrow \text{BS}$
P_{41}	$v_4 \rightarrow v_{16} \rightarrow v_{17} \rightarrow \text{BS}$
P_{42}	$v_4 \rightarrow v_8 \rightarrow v_{17} \rightarrow \text{BS}$

to $v_3, v_4, v_7, v_8, v_{11}, v_{12}, v_{16}$ and v_{17} ; the information in each node is shown in Table 5.

In the data forwarding phase, when v_4 obtains p_{40} as the feedback path, it will check its local memory and find that v_3 is the next hop of v_4 in p_{40} , and v_4 sends the ready packets to v_3 . Likewise, v_3 will find v_7 is the next hop; the packet will be sent to v_7 . v_7 finds that BS exists in p_{40} so that the packet will be sent to the BS.

When the source node has data to send, it sends a request to the BS in the data forwarding phase. After receiving the request, the BS determines the optimal path according to the global energy information, the distribution of all nodes, and each node's relaying pressure. Then, the BS feeds the ID set of the path back to the source node. When there is more than one path ID in the set, the source node first checks whether the matched next hop exists with the last ID in the local memory; if it exists, the packet will be sent to the next hop. Otherwise, the node will check the previous ID in the set.

TABLE 5: The feedback information.

Node	The next hop and path ID
v_3	$v_7, \{P_{40}\}$
v_4	$v_3, \{P_{40}\}$
v_7	$v_{16}, \{P_{41}\}$
v_8	$\text{BS}, \{P_{40}\}$
v_{16}	$v_{17}, \{P_{42}\}$
v_{17}	$v_{17}, \{P_{41}\}$
v_{17}	$\text{BS}, \{P_{41}, P_{42}\}$

Specifically, the data forwarding phase includes two steps:

Step 1: global energy information cookie: the global energy information is important to decide the routing for the source node in GGNRP. The BS maintains the global energy information in GGNRP. The real-time energy information can be used to make the routing decision precisely. However, it will cause a great deal of energy consumption. GGNRP adopts the energy information cookie mechanism to avoid additional power consumption. When the source node starts to transmit data, the BS will compute every node's energy cost and update each node's energy information according to the number of packets, the size of each packet, and the distribution of the network.

Step 2: routing decision: the routing decision of GGNRP is made by the BS. The proposed routing decision algorithm is shown as Algorithm 1.

The BS determines the optimal routing according to the global energy information, the distribution of all nodes, and each node's relaying pressure. The lower the maximum relaying pressure of the path is and the less energy the path costs, the more likely the path is to be optimal. Since the multiple paths stored by each source node at the BS are the minimum number of hops, its performance in terms of delay is particularly excellent. However, the low-energy and high-burden nodes in the network cannot be well protected due to the limited paths. The BS will simulate the process of the nodes communicating with the BS. In the simulation, every time the packet reaches a node, the node will compare itself with its neighbors who have higher residual energy. If there are no nodes that have a higher weight than the current node, the packet will be sent to the original next hop. Otherwise, the packet will be sent to the node that has the highest weight and the BS will record the new path ID. This process repeats until the packet reaches the BS. Finally, the BS feeds the set of path IDs back to the source node. The packet is sent according to the path IDs set. When there is more than one ID in the set, the source node will check whether the next hop exists in the local memory in reverse order; if it exists, the packet will be sent to the next hop. Otherwise, the node will check the previous ID in the set in the memory.

GGNRP protects the lower-residual-energy nodes and higher-burden nodes by using the node relaying pressure as one of the routing decision factors. It reduces energy consumption by setting the path cost as one of the routing factors. In addition, all routing paths in GGNRP are almost the shortest

```

Input: Pathi
Output: S, w
(1) For pij ∈ Pathi
(2)   For vk ∈ pij
(3)     If maxij < Pressvk
(4)       maxij = Pressvk
(5)     End If
(6)   End For
(7)   If w < (1 / (maxij * costij))
(8)     w = (1 / (maxij * costij))
(9)     S = pij
(10)   End If
(11) End For
(12) Return S, w

```

ALGORITHM 1: Routing decision algorithm.

paths to the BS, which decrease the transmission delay and provide more transmission opportunities in VNDN.

5. Simulation Results

5.1. Simulation Setup. To validate the effectiveness of the proposed GGNRP, the performance of GGNRP is evaluated and compared with FAF-EBRM [37] in terms of energy consumption and delay. The power consumption and average delay of different schemes with a varying number of nodes are considered in different scenarios. The setup of simulation parameters is shown in Table 6.

5.2. Power Consumption and Average Delay. In this scenario, the monitoring area is set to $1000 \times 1000 \text{ m}^2$, and the number of nodes is set to 100. The lowest residual energy and average delay of GGNRP and FAF-EBRM are shown in Figures 7 and 8, respectively.

When the communication begins, the initial energy of all the nodes is 10 mJ. Before the 21st round, FAF-EBRM's lowest energy is slightly higher than that of GGNRP, which means that FAF-EBRM has a better performance than GGNRP in terms of energy consumption with low load. The reason is that FAF-EBRM chooses the high-energy-density forwarding area as the transmitting direction and GGNRP chooses the path cost and the node relaying pressure as the routing decision factors. In the 20th round, the energy of FAF-EBRM is $3 \mu\text{J}$ higher than that of GGNRP. However, in the 21st round of transmission, the energy of GGNRP is $263 \mu\text{J}$ higher than that of FAF-EBRM. Over time, the difference between GGNRP and FAF-EBRM increases. In the 57th round, the lowest residual energy of FAF-EBRM reaches 0 due to the first node, which consumes its energy, while GGNRP still has $1322 \mu\text{J}$ of energy. These results illustrate that choosing the path cost and the node relaying pressure as the routing decision metrics has an advantage over choosing the high-energy-density forwarding area as the transmitting direction. This is because FAF-EBRM does not consider the total communication cost in the routing decision. GGNRP can support 66 rounds of communications, which indicates that GGNRP has 15.8% greater efficiency than FAF-EBRM.

As shown in Figure 8, in the first round, the average delays of FAF-EBRM and GGNRP are 34.1 ms and 14.4 ms, respectively, which means that the average delay of GGNRP is 57.8% lower than that of FAF-EBRM. From an overall perspective, the average delay of FAF-EBRM varies around 38.0 ms. While GGNRP maintains an average delay of 14 ms and has little delay variation. In the 24th communication round, the difference between FAF-EBRM and GGNRP reaches its maximum value, and the average delays of FAF-EBRM and GGNRP are 61.6 ms and 13.6 ms, respectively. The former is 4.53 times greater than the latter. In the 30th round of communication, the difference between FAF-EBRM and GGNRP reaches the minimum value, and the average delays of FAF-EBRM and GGNRP are 30.3 ms and 13 ms, respectively. The delay of FAF-EBRM is still 2.33 times that of GGNRP. For FAF-EBRM, nodes have the opportunity to be the next hop as long as their locations relative to the base station are closer than that of the source node. When the source node or the energy distribution is different, the delay of the network is different, which results in large delay variation in the network. In addition, FAF-EBRM prefers to the nodes in the energy density area as its next hop, which leads to packets being forwarded frequently among nodes and results in a higher delay than GGNRP. For GGNRP, the routing paths are optimized, GGNRP maintains its delay at a low level, and its delay variation is small.

5.3. Performance Comparisons at Different Scales. To evaluate the adaptability of GGNRP and FAF-EBRM at different scales, the power consumption and average delay results are shown in Figures 9 and 10, respectively, with varying numbers of nodes.

As shown in Figure 9, the remaining power is expressed as the number of forwarding packets. It is obvious that GGNRP can always afford more packets than FAF-EBRM, and the difference between them grows larger as the number of nodes increases. When the number of nodes is 150, the performance of GGNRP is almost the same as that of FAF-EBRM, and the numbers of forwarding packets of GGNRP and FAF-EBRM are 8044 and 7651, respectively. The performance of GGNRP is 5.13% higher than that of FAF-EBRM. When the number of nodes is 350, GGNRP can

TABLE 6: The simulation parameters setup.

Parameter	Value
The initial energy	10 mJ
The size of packet	1000 b
E_{amp}	10 pJ/b/m ²
E_{elec}	10 nJ/b
r	10 m

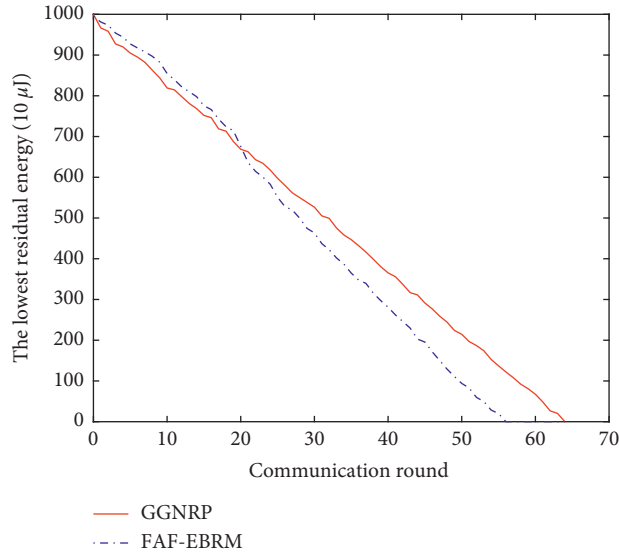


FIGURE 7: Power consumption of different schemes.

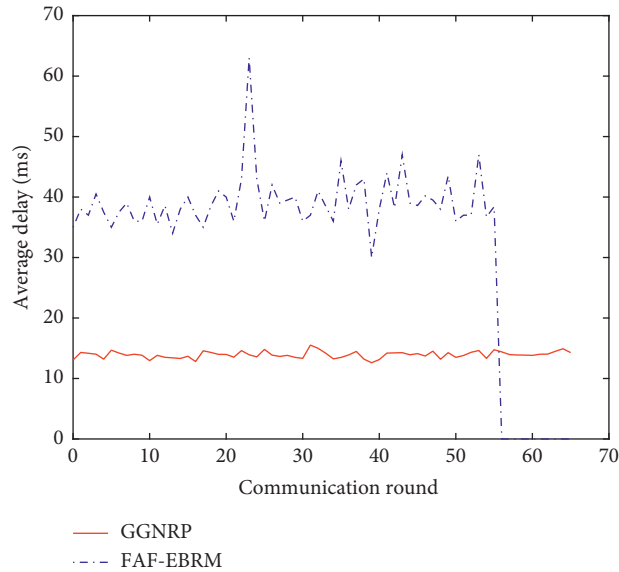


FIGURE 8: Average delay of different schemes.

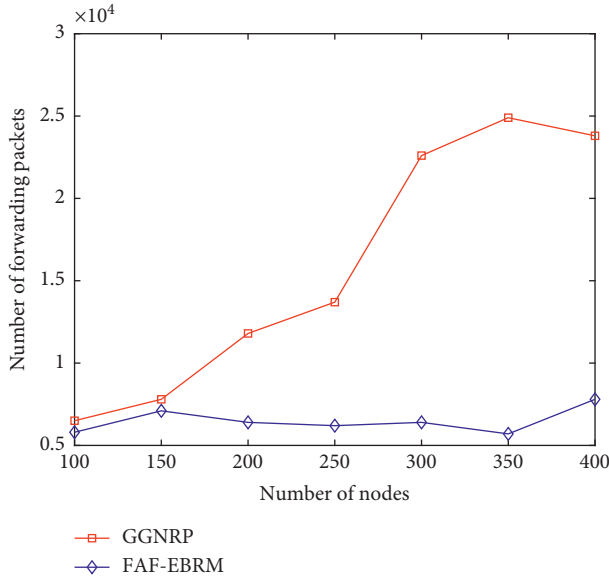


FIGURE 9: Power consumption of different schemes with varying number of nodes.

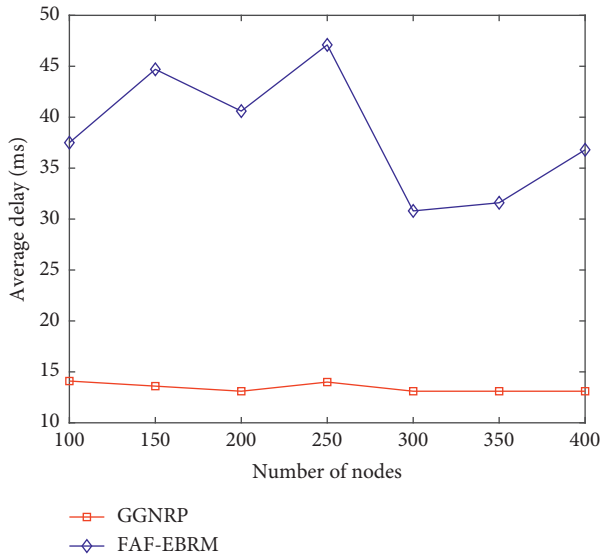


FIGURE 10: Average delay of different schemes with varying number of nodes.

afford 25032 packets, which is 4.37 times that of FAF-EBRM. These results show that GGNRP is more scalable than FAF-EBRM as the number of nodes varies. The power consumption of FAF-EBRM may even increase as the scale of WSNs grows. This occurs because a packet in FAF-EBRM needs to be transmitted to more nodes in the energy density area.

As shown in Figure 10, the average delay of GGNRP remains almost constant at approximately 13.7 ms, while the average delay of FAF-EBRM clearly varies. The average delay of FAF-EBRM is much larger than that of GGNRP. When the number of nodes is 100, the average delay of GGNRP is

13.98 ms, which is only 36.8% of that of FAF-EBRM. The results indicate that the proposed GGNRP decreases the average delay by 63.2%. When the number of nodes is 250, the average delays of GGNRP and FAF-EBRM are 13.934 ms and 46.1762 ms, respectively, and the difference between FAF-EBRM and GGNRP reaches the maximum value. When the number of nodes is 300, the average delays of GGNRP and FAF-EBRM are 13.13 ms and 29.91 ms, respectively, and the difference between FAF-EBRM and GGNRP reaches the maximum value. The results show that GGNRP is effective with varying scales.

5.4. Security Analysis. The security analysis of the proposed scheme is summarized as follows.

5.4.1. Distribution. Instead of setting a root key, the proposed key management scheme utilizes the distributed blockchain as a trust anchor to guarantee the authenticity of the keys by storing a hash value. The failure of a single node does not affect the key acquisition and verification, which avoids the failure of single point.

5.4.2. Trustworthiness. Blockchain solves the problem of cross-domain key authentication. In the absence of a root key, interdomain nodes cannot verify the legitimacy of the public key, which results in trust crisis. The proposed key management scheme stores the hash value of the public key in the blockchain and solves the trust problem of the public key by the tamper-evident nature of the blockchain.

6. Conclusion

In this article, a blockchain-based key management and green routing scheme is proposed for VNDN. A key management scheme is presented based on the blockchain by taking advantage of the distributed and antitampering characteristics of the blockchain. In this scheme, a flat hierarchical structure reduces the number of signatures and identity verifications needed to safely and efficiently verify the legitimacy of the producer. We elaborated on the mechanism of the scheme and the characteristics of its realization. To decrease the power consumption of nodes close to the BS and the transmission delay, the metric of node relay pressure is introduced, which makes the nodes with lower relay pressure more likely to be members of the selected forwarding path. The proposed GGNRP uses route exploration and feedback mechanisms in the routing establishment phase to avoid the coverage hole problem. In addition, GGNRP uses the node relay pressure and energy consumption as metrics in routing decisions, which reduces power consumption and transmission delays. The simulation results show that GGNRP can achieve better performance than FAF-EBRM in terms of power consumption and average delay.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon reasonable request.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this article.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (nos. 61772562 and 62062019), the Key Project of Hubei Provincial Science and Technology Innovation Foundation of China (no. 2018ABB1485), and the Youth Elite Project of State Ethnic Affairs Commission of China (no. 2016-3-08).

References

- [1] S. Guo, X. Hu, S. Guo, X. Qiu, and F. Qi, "Blockchain meets edge computing: a distributed and trusted authentication system," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 3, pp. 1972–1983, 2020.
- [2] L. Yao, A. Chen, J. Deng, J. Wang, and G. Wu, "A cooperative caching scheme based on mobility prediction in vehicular content centric networks," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 6, pp. 5435–5444, 2018.
- [3] C. Chen, J. Hu, T. Qiu, M. Atiquzzaman, and Z. Ren, "CVCG: cooperative V2V-aided transmission scheme based on coalitional game for popular content distribution in vehicular Ad-Hoc networks," *IEEE Transactions on Mobile Computing*, vol. 18, no. 12, pp. 2811–2828, 2019.
- [4] L. Silva, N. Magaia, B. Sousa et al., "Computing paradigms in emerging vehicular environments: a review," *IEEE/CAA Journal of Automatica Sinica*, vol. 8, no. 3, pp. 491–511, 2021.
- [5] C. Chen, C. Wang, T. Qiu, N. Lv, and Q. Pei, "A secure content sharing scheme based on blockchain in vehicular named data networks," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 5, pp. 3278–3289, 2020.
- [6] S. H. Ahmed, S. H. Bouk, D. Kim, D. B. Rawat, and H. Song, "Named data networking for software defined vehicular networks," *IEEE Communications Magazine*, vol. 55, no. 8, pp. 60–66, 2017.
- [7] T. Jin, X. Zhang, Y. Liu, and K. Lei, "BlockNDN: a bitcoin blockchain decentralized system over named data networking," in *Proceedings of the 2017 9th International Conference on Ubiquitous and Future Networks (ICUFN)*, pp. 75–80, Milan, Italy, 2017.
- [8] J. Guo, M. Wang, B. Chen, S. Yu, H. Zhang, and Y. Zhang, "Enabling blockchain applications over named data networking," in *Proceedings of the 2019 IEEE International Conference on Communications (ICC)*, pp. 1–6, Shanghai, China, 2019.
- [9] E. G. AbdAllah, H. S. Hassanein, and M. Zulkernine, "A survey of security attacks in information-centric networking," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 3, pp. 1441–1454, 2015.
- [10] D. B. Rawat, R. Doku, A. Adebayo, C. Bajracharya, and C. Kamhoua, "Blockchain enabled named data networking for secure vehicle-to-everything communications," *IEEE Network*, vol. 34, no. 5, pp. 185–189, 2020.
- [11] J. Lou, Q. Zhang, Z. Qi, and K. Lei, "A blockchain-based key management scheme for named data networking," in *Proceedings of the 2018 1st IEEE International Conference on Hot Information-Centric Networking (HotICN)*, pp. 141–146, Shenzhen, China, 2018.
- [12] F. Ahmad, C. A. Kerrache, F. Kurugollu, and R. Hussain, "Realization of blockchain in named data networking-based internet-of-vehicles," *IT Professional*, vol. 21, no. 4, pp. 41–47, 2019.
- [13] C. Chen, J. Li, V. Balasubramaniam, Y. Wu, Y. Zhang, and S. Wan, "Contention resolution in Wi-Fi 6-enabled internet of things based on deep learning," *IEEE Internet of Things Journal*, vol. 8, no. 7, pp. 5309–5320, 2021.
- [14] C. Chen, B. Liu, S. Wan, P. Qiao, and Q. Pei, "An edge traffic flow detection scheme based on deep learning in an intelligent transportation system," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 3, pp. 1840–1852, 2021.
- [15] K. Harish, A. Harneet, and R. K. Singla, "Energy-aware fisheye routing (EA-FSR) algorithm for wireless mobile sensor networks," *Egyptian Informatics Journal*, vol. 14, no. 1, pp. 235–238, 2013.
- [16] A. H. Sodhro, J. J. P. C. Rodrigues, S. Pirbhulal, N. Zahid, A. R. L. de Macedo, and V. H. C. de Albuquerque, "Link optimization in software defined IoV driven autonomous transportation system," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 6, pp. 3511–3520, 2021.
- [17] G. Sedky and A. E. Mougny, "BCXP: blockchain-centric network layer for efficient transaction and block exchange over name data networking," in *Proceedings of the 2018 IEEE 43rd Conference on Local Computer Networks (LCN)*, pp. 449–452, Chicago, IL, USA, 2018.
- [18] C. Chen, Y. Zhang, Z. Wang, S. Wan, and Q. Pei, "Distributed computation offloading method based on deep reinforcement learning in ICV," *Applied Soft Computing*, vol. 103, Article ID 107108, 2021.
- [19] J. Ma, T. Li, J. Cui, Z. Ying, and J. Cheng, "Attribute-based secure announcement sharing among vehicles using blockchain," *IEEE Internet of Things Journal*, vol. 8, no. 13, pp. 10873–10883, 2021.
- [20] T. Chatterjee, S. Ruj, and S. D. Bit, "Security issues in named data networks," *Computer*, vol. 51, no. 1, pp. 66–75, 2018.
- [21] T. Song, B. Cui, R. Li, J. Liu, and J. Shi, "Smart contract-based trusted content retrieval mechanism for NDN," *IEEE Access*, vol. 8, pp. 85813–85825, 2020.
- [22] H.-K. Yang, H.-J. Cha, and Y.-J. Song, "Secure identifier management based on blockchain technology in NDN environment," *IEEE Access*, vol. 7, pp. 6262–6268, 2019.
- [23] B. Li, M. Ma, and R. Xia, "Hierarchical identity-based security mechanism using blockchain in named data networking," in *Proceedings of the 2020 3rd International Conference on Hot Information-Centric Networking (HotICN)*, pp. 148–153, Hefei, China, 2020.
- [24] G. Mauri, M. Gerla, F. Bruno, M. Cesana, and G. Verticale, "Optimal content prefetching in NDN vehicle-to-infrastructure scenario," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 3, pp. 2513–2525, 2017.
- [25] M. F. Majeed, S. H. Ahmed, and M. N. Dailey, "Enabling push-based critical data forwarding in vehicular named data networks," *IEEE Communications Letters*, vol. 21, no. 4, pp. 873–876, 2017.
- [26] J. Cui, Y. Wang, J. Zhang, Y. Xu, and H. Zhong, "Full session key agreement scheme based on chaotic map in vehicular Ad-

- Hoc networks,” *IEEE Transactions on Vehicular Technology*, vol. 69, no. 8, pp. 8914–8924, 2020.
- [27] J. Zhang, J. Cui, H. Zhong, Z. Chen, and L. Liu, “PA-CRT: Chinese remainder theorem based conditional privacy-preserving authentication scheme in vehicular Ad-Hoc networks,” *IEEE Transactions on Dependable and Secure Computing*, vol. 18, no. 2, pp. 722–735, 2021.
- [28] S. Hameed, S. A. Shah, Q. S. Saeed et al., “A scalable key and trust management solution for IoT sensors using SDN and blockchain technology,” *IEEE Sensors Journal*, vol. 21, no. 6, pp. 8716–8733, 2021.
- [29] W. Zheng, K. Wang, and F.-Y. Wang, “GAN-based key secret-sharing scheme in blockchain,” *IEEE Transactions on Cybernetics*, vol. 51, no. 1, pp. 393–404, 2021.
- [30] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, “Energy-efficient communication protocol for wireless microsensor networks,” in *Proceedings of the 33rd Annual Hawaii International Conference on System Sciences*, Maui, HI, USA, 2000.
- [31] S. Lindsey and C. S. Raghavendra, “PEGASIS: power-efficient gathering in sensor information systems,” in *Proceedings of the 2002 IEEE Aerospace Conference*, Big Sky, MT, USA, 2002.
- [32] M. Park, J. Choi, Y. Han, and T. Chung, “An energy efficient concentric clustering scheme in wireless sensor networks,” in *Proceedings of the 5th International Joint Conference on INC, IMS and IDC*, pp. 58–61, Seoul, Republic of Korea, 2009.
- [33] C. Li, L. Wang, T. Sun et al., “Topology analysis of wireless sensor networks based on nodes’ spatial distribution,” *IEEE Transactions on Wireless Communications*, vol. 13, no. 5, pp. 2454–2453, 2014.
- [34] C. Zhu, L. T. Yang, L. Shu, J. J. P. C. Rodrigues, and T. Hara, “A geographic routing oriented sleep scheduling algorithm in duty-cycled sensor networks,” in *Proceedings of the 2012 IEEE International Conference on Communications (ICC)*, pp. 5473–5477, Ottawa, Canada, June 2012.
- [35] F. Ren, J. Zhang, T. He, C. Lin, and S. K. D. Ren, “EBRP: energy-balanced routing protocol for data gathering in wireless sensor networks,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 22, no. 12, pp. 2108–2125, 2011.
- [36] J. Luo, J. Hu, D. Wu, and R. Li, “Opportunistic routing algorithm for relay node selection in wireless sensor networks,” *IEEE Transactions on Industrial Informatics*, vol. 11, no. 1, pp. 112–121, 2015.
- [37] D. Zhang, G. Li, K. Zheng, X. Ming, and Z.-H. Pan, “An energy-balanced routing method based on forward-aware factor for wireless sensor networks,” *IEEE Transactions on Industrial Informatics*, vol. 10, no. 1, pp. 766–773, 2014.

Research Article

Classification of Abnormal Traffic in Smart Grids Based on GACNN and Data Statistical Analysis

F. F. Hu,¹ S. T. Zhang,¹ X. B. Lin,¹ L. Wu,¹ and N. D. Liao ²

¹CSG Power, Dispatching Control Center, Guangzhou 510663, China

²Hunan Provincial Engineering Research Center of Electric Transportation and Smart Distribution Network (Changsha University of Science and Technology), Changsha 410114, China

Correspondence should be addressed to N. D. Liao; Indy97@csust.edu.cn

Received 11 March 2021; Accepted 7 June 2021; Published 17 June 2021

Academic Editor: Ricardo Chaves

Copyright © 2021 F. F. Hu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

With the continuous development of smart grids, communication networks carry more and more power services, and at the same time, they are also facing more and more security issues. For example, some malicious software usually uses encryption technology or tunnel technology to bypass firewalls, intrusion detection systems, etc., thereby posing a serious threat to the information security of smart grids. At present, the classification of network traffic mainly depends on the correct extraction of network protocol characteristics. However, the process of extracting network features by some traditional methods is time-consuming and overly dependent on experience. In order to solve the problem of accurate classification of power network traffic, this paper proposes a method of convolutional neural network based on genetic algorithm optimization (GACNN) and data statistical analysis. This method can simultaneously extract the time characteristics between different packet groups and the spatial characteristics in the same packet group. Therefore, it greatly saves manpower and gets rid of the dependence on experience value. The proposed method has been tested and verified on the UNSW-NB15 dataset and the real dataset collected by the power company. The results show that the proposed method can correctly classify abnormal network flows and is much better than traditional machine learning methods. In large-scale real network flow scenarios, the detection rate of the proposed method exceeds 97%, while the traditional method is generally less than 90%.

1. Introduction

1.1. Background. As a next-generation grid, smart grids have the characteristics of high controllability, high energy efficiency, and self-healing. Smart grids have been rapidly built in many countries and regions in the world, providing great convenience to people's lives [1]. Compared with traditional power grid, smart grids require more monitoring and control devices and are more widely distributed. In order to achieve comprehensive and real-time monitoring, low-cost wireless communication network and widely distributed public Internet are increasingly used in power grid. However, the massive access of the public network in the power system provides more access for malicious attacks. This will bring more harm to the power grid and users [2]. With some new energy access to smart grids, the security risk of smart

grids increases greatly, such as violent attacks, denial of service attacks, and computer viruses [3].

Due to different business requirements in different areas of smart grids, the requirements for underlying network communication are also different. Table 1 shows the main network technology architecture of the current smart grids [4]. At present, the network communication of smart grids mainly depends on advanced metering infrastructure (AMI) [5]. The AMI network is mainly composed of home local area network (HAN), neighborhood network (NAN), and wide area network (WAN) [6].

It can be seen from Table 1 that different power users have different demands on the network. Timely monitoring and analysis of network traffic is of great significance to improving network performance and security defense. When the network traffic monitoring system is applied to the

TABLE 1: Network architecture of smart grids.

Network type	Network technique	Application scenarios
Wide area network (WAN)	IP, DWDM	Provide power data network (backbone network), Internet interconnection, and routing functions
	MPLS\MPLS VPN	Label switching is provided in backbone network to isolate traffic of different services
Access network (AN)	ATM	Asynchronous Transfer Mode
	SDH	Provides physical access to MAN and WAN
	MSTP	It can be used for LAN (Ethernet) access to MAN
	GPRS\3G\4G\5G	Access WAN through mobile communication network
Local area network (LAN)	PON	It is a typical passive optical fiber network
	IEEE 802.3\802.1q	LAN of electric power enterprise
	RS-485, PROFIBUS and other traditional field buses	Production control fields such as power plants and substations
	Industrial Ethernet	Interconnection of IED equipment in production control fields such as power plants and substations
Field area network (FAN)	N-PLC, B-PLC/BPL (narrowband, broadband power line carrier communication)	Used for data transmission such as metering and instrument data collection
	Wireless sensor networks	Data acquisition, monitoring and monitoring of power transmission, distribution and consumption side
	Internet of things, RFID	Collection of label data in equipment inspection
	PON/EPON/FTTH	Intelligent residential area provides optical fiber access for home users
Home area network (HAN)	N-PLC\B-PLC\BPL	Provide local network and home network access, remote meter reading and Internet access
	WLAN 802.11	Remote meter reading with local network or home network access
	Wireless sensor network	Used for control of smart home and home appliances in HAN

power enterprise, it can not only monitor the network traffic in real time but also analyze and warn the abnormal situation of the network [7].

Some early network traffic monitoring models obtain network characteristic information such as bandwidth and link utilization and judge whether network abnormalities occur according to thresholds. Although these models have certain traffic alarm functions, the performance of the models mainly depends on the accuracy of the characteristic values, and the model thresholds are generally set manually [8].

With the rapid development of next-generation networks, integrated services such as voice, video, and data are running in parallel, and the proportion of new traffic types P2P, streaming media, and games continues to increase. Today's network traffic recognition technology has some new problems:

- (1) The detection efficiency of the existing methods is low, especially in the high-speed online traffic classification, and the detection method has a large storage overhead and a large amount of calculation.
- (2) Some new network services mostly use variable ports, data encryption, and dedicated protocols. Existing methods can no longer accurately extract the characteristics of network traffics, which affects the recognition accuracy.

At present, as a large number of different AMI devices are connected to the network, it provides detailed basic data in many aspects for the power system, but at the same time, it

also brings more network security risks. How to grasp the characteristics of different network behaviors in time and better understand the state of network traffic has become one of the key core issues in power communication research.

1.2. Our Contributions. This paper first establishes a standard power network flow metadata to eliminate the problem of multisource and heterogeneous equipment information service coordination. (1) Through the real-time or offline collection of the entire multisource network equipment flow, the unified analysis of different equipment, different network levels, and mass flow data is realized. (2) Through statistical analysis of standardized flow metadata, network congestion or abnormal conditions can be found as early as possible, which provides data support for power network security situation assessment.

Secondly, unlike traditional machine learning methods, this paper uses deep learning to analyze network traffic metadata and quickly find out the temporal and spatial characteristics of traffic. The main advantages of this method are as follows: (1) it is not necessary to judge the importance of features in advance but directly input the metadata to the model for training after preprocessing. (2) The model has self-learning ability, which eliminates the problem that the accuracy of traditional methods is excessively dependent on expert experience or threshold setting. (3) The model has the ability of dealing with encryption and multilevel business traffic, which overcomes the problem that traditional methods cannot analyze network traffic content.

1.3. Organization. The rest of this paper is organized as follows: Section 2 summarizes the methods of network traffic classification. Section 3 introduces the convolutional neural network (CNN), genetic algorithm (GA), and network traffic collection and preprocessing. Section 4 introduces the GACNN method proposed in this paper in detail. Section 5 details the process and results of two experiments conducted on the UNSW-NB15 and real power flow datasets. Finally, Section 6 summarizes the main work of the paper and the future research direction.

2. Related Work

With the construction of smart grids, power data networks and the business systems carried by them have developed rapidly, and a large amount of network traffic is generated in these systems every day.

Power critical information infrastructure such as AMI is the nerve center of economic and social operations, and it is also an important target that may be attacked through the network. A large number of experiments have found that the power network traffic under attack will be abnormal. These kinds of abnormal traffic are usually caused by malicious network attacks (such as worm propagation, DDOS attacks, botnets, and viruses) or network configuration errors and occasional line interruptions [9].

As one of the key technologies of network management and network security, network traffic classification can not only optimize network configuration and reduce network security risks but also provide better service quality. In the past two decades, domestic and foreign researchers have carried out a series of related studies and achieved some outstanding results.

Initially, the network traffic classification technology is relatively simple, because different network applications use different port numbers for communication, so the port number can be used as a function to identify network traffic [10]. Wang et al. [11] proposed a traffic detection algorithm based on port scanning behavior. The algorithm determines whether there is an abnormality in network traffic based on the ratio and similarity of the number of hosts and ports. However, some network protocols currently do not use fixed ports for communication, so this method cannot cope with the problem of sudden changes in ports.

In order to accurately identify different network services on dynamic ports, some researchers have proposed network traffic identification methods based on payload characteristics, such as deep packet inspection (DPI) [12, 13] and deep flow inspection (DFI) [14, 15].

The DPI technology mainly determines the type of each network service by judging whether the network characteristics match the fingerprint library characteristics. Sun et al. [12] first used DPI technology to quickly and accurately preclassify the traffic and then calculated the random characteristics of the packet load, so as to achieve effective identification of encrypted traffic.

DFI mainly uses application identification technology based on traffic behavior. This method believes that different applications will present different states on network session

connections or data streams, but DFI does not pay attention to application load. In order to achieve the purpose of lightweight protection of the power Internet of Things, Wang and Wei [14] used DFI technology to analyze the collected network traffic of the Internet of Things and formed a feasible security prevention and control strategy model.

The DPI and DFI methods to process network traffic mainly rely on the application layer message characteristics of the service, and some of them need to analyze the message content, which may infringe user privacy. In addition, with the continuous emergence of new services, the feature database must be constantly updated so that the original method may be effective, which requires a lot of work.

A new generation of traffic detection methods based on statistical characteristics came into being. This classification method relies on statistical features or time series features and can handle encrypted and unencrypted traffic. These methods usually use classic machine learning algorithms to process analysis [16].

In recent years, machine learning methods have become very popular and have been widely used in fields such as image, sound, and text processing [17–25]. The traditional machine learning methods used in network traffic recognition mainly include Bayesian algorithms, support vector machine (SVM) algorithms, decision tree and integrated learning algorithms, etc. [26]. Distributed denial of service (DDoS) has always been a serious threat to the Internet. Hou et al. [27] proposed a scheme of feature selection and machine learning to identify DDoS traffic. The frequency of malicious activities such as botnets and port scanning is increasing. Although these attacks are simple, they may allow unauthorized network access. Flanagan et al. [28] proposed an MCODET anomaly detection system, which mainly uses the polynomial regression technique of clustering density to detect the abnormal behavior of NetFlow data.

In the power network anomaly detection method, some machine learning methods have also begun to be applied to this area of research.

Fei et al. proposed an improved CUSUM detection algorithm (BF-DT-CUSUM) that dynamically updates the threshold of address statistics, which is used to detect distributed denial of service attacks in power industrial control systems. Simulation experiments verify that the algorithm has good speed and accuracy in response to DDoS attacks [29]. However, the algorithm is difficult to detect other unknown attack types.

In order to more effectively classify the increasing traffic of the power business system and to improve the business processing speed of the power system, Xu proposed a real-time traffic classification method for power business based on an improved random forest algorithm [30]. This method improves the real-time classification by pruning the random forest based on the classification interval weighting. Du et al. analyzed the flow data structure of the power network and verified that the normal flow data of the power has stable information entropy. On this basis, an algorithm based on five-tuple entropy of traffic and SVM is proposed to identify

abnormal traffic [31]. Recent research on the use of machine learning algorithms for traffic classification is also mainly focused on the selection of optimized features [32].

Most of these machine learning-based traffic classification methods rely on feature selection, which limits their generalizability.

Recently, deep learning has been well applied in image recognition, natural language processing, and sentiment analysis. These methods can automatically select features through the training process and have strong versatility.

There are three main methods for detecting network traffic anomalies based on deep learning: deep Boltzmann machine [33, 34], stacked autoencoder [35, 36], and CNN [16, 37]. Ertam and Avci developed GA-WKELM software. This method is based on the combination of genetic algorithm and extreme learning machine and mainly solves the problem of parameter optimization and selection of deep neural networks. However, this method can only complete one training and cannot dynamically update parameters and training samples [38]. Wang et al. proposed an end-to-end traffic classification model. This model processes the network traffic data into a specific file format and then classifies the network traffic through a one-dimensional CNN [39].

Due to business security considerations, the power data dispatching network uses all established channels exclusively and must not be reused. Gao and Yao [40] believe that the statistical characteristics of power communication network traffic have self-similarity, multifractal, periodicity, chaos, and other characteristics. The bandwidth occupied by data traffic of the power data dispatch network can only be a rigid superposition of the business traffic carried. Based on the above characteristics of power network flow, Lv and others used the autoregressive moving average (ARMA) model to predict the power network flow [41]. Lin et al. established a power grid operation situation awareness model based on the fuzzy analytic hierarchy process and the LSTM-attention mechanism [42].

Compared with traditional machine learning methods, deep Boltzmann machine-based methods can extract high-dimensional features of traffic data through learning. However, the robustness of this method is poor. When the input data contain noise, the extracted features will be inaccurate.

The anomaly detection method based on stacked autoencoders can learn traffic data layer by layer and extract traffic features with high accuracy. However, when the traffic data are destroyed, the detection accuracy of this method will be reduced.

The method based on CNN has strong robustness and high detection performance. However, the traditional CNN method generally uses a gradient descent algorithm for training, and if the initial weight of the network is incorrectly selected, it will also affect the learning performance and make the model fall into a local optimal state.

Through the analysis of the advantages and disadvantages of existing deep learning methods in network traffic classification, this paper mainly selects the CNN model to extract the characteristics of power network traffic by itself, eliminating human intervention or relying on expert

knowledge. In addition, in order to improve the problem of CNN model parameter selection, considering that the genetic algorithm has an effective search ability for global and local optimal solutions, the genetic algorithm is used to find optimal solutions for CNN model parameters.

3. Background

3.1. Introduction to CNN. CNN was originally proposed by LeCun et al. in 1989 to solve the problem of digital image recognition [43]. CNN is also a neural network specially used to process data with a known grid-like topology. For example, time series data can be regarded as a one-dimensional grid sampled at a certain time interval, and image data can be regarded as a two-dimensional grid composed of pixels. In the calculation, the network mainly uses a mathematical operation called convolution. Convolution is a special linear operation, which can replace the general matrix calculation to achieve multiple operation effects [44]. With the development of CNN, many variants of convolution network structure appear, but their basic structures are mostly similar, mainly including input layer, convolution layer, pooling layer, full connection layer, and output layer. Figure 1 shows the basic structure of CNN network.

The input layer is used to input data or images. In order to facilitate the calculation of convolution layer, the input data needs to be preprocessed.

Convolution layer and convolution kernel of convolution layer are mainly used for feature extraction of input information, and the convolution function is shown in the following formula:

$$x_j^n = f \left(\sum_{i \in D_j} x_i^{n-1} K_{ij}^n + b_j^n \right), \quad (1)$$

where D_j is the input characteristic data, x_j^n is the characteristic value J of the n th layer, K_{ij}^n is convolution kernel function, $f()$ is the activation function, and b_j^n is the bias parameter. The activation functions used in this paper are sigmoid, ReLU, and softmax, and their formulas are, respectively,

$$S(x) = \frac{1}{1 + e^{-x}}, \quad (2)$$

$$\text{ReLU}(x) = \begin{cases} x, & \text{if } x > 0, \\ 0, & \text{if } x \leq 0, \end{cases} \quad (3)$$

$$\sigma(z)_j = \frac{e^{z_j}}{\sum_{k=1}^K e^{z_k}}. \quad (4)$$

The convolutional layer and the pooling layer are calculated alternately. The calculation of the pooling layer is

$$x_j^{n+1} = f \left(\sum_j x_j^n \omega_j^{n+1} + b_j^{n+1} \right), \quad (5)$$

where ω_j^{n+1} is the weight constant of the feature map of the pooling layer.

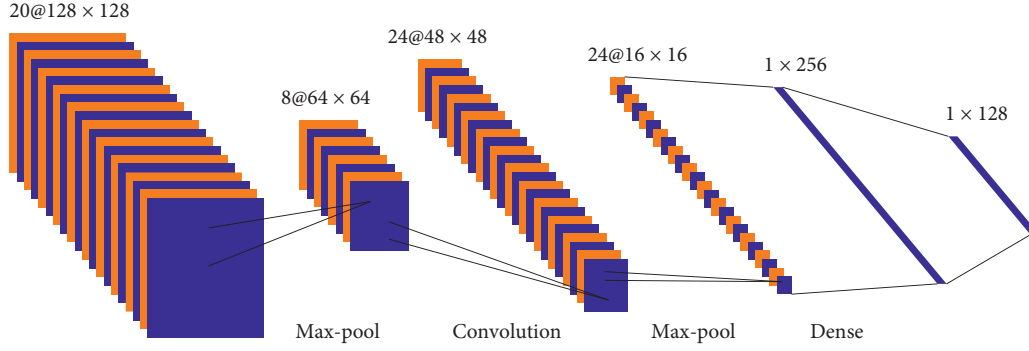


FIGURE 1: The basic structure of CNN.

Before the output layer gets the result, in the n -layer convolutional neural, as the input sample, f_n represents the activation function of each layer pooling, and ω^n represents the connection weight of each layer. The calculation of this process can be expressed as

$$y = f_n(\dots(f_2(f_1(x \cdot \omega^1)\omega^2))\dots)\omega^n. \quad (6)$$

The result of forward propagation is calculated, and the error is compared with the known label value. The error function is expressed as

$$E = \frac{1}{n} \sum_{i=1}^N \sum_{j=1}^M (y'_{ji} - y_{ji})^2, \quad (7)$$

where N is the number of training samples, M is the number of output neurons, y'_{ji} is the expected output value of the j -th output node of the i -th sample, and y_{ji} is the actual output value of the j -th output node of the i -th sample.

3.2. Genetic Algorithm (GA). GA was first proposed by John Holland in 1962. The algorithm is designed and proposed according to the laws of biological evolution in nature. It is a computational model used to simulate the natural selection and genetic mechanism of Darwin's theory of biological evolution. As a metaheuristic search strategy, the GA is mainly used to find the best super parameter algorithm of machine learning [45]. Generally speaking, the genetic algorithm is divided into five stages [46]:

- (1) Initial population: set the evolution algebra counter $t=0$, set the maximum evolution algebra T , and randomly generate m individuals as the initial population $P(0)$.
- (2) Fitness function: the fitness of each individual in population $P(T)$ was calculated.
- (3) Selection: the selection operator is applied to the population. The purpose of selection is to inherit the optimized individual directly to the next generation or to generate new individuals through pairing and crossover and then inherit the next generation. The selection operation is based on the assessment of the fitness of the individuals in the population.

- (4) Crossover: the crossover operator is applied to the population. Crossover operator plays a key role in the genetic algorithm.
- (5) Mutation: the mutation operator is applied to the population. That is to change the gene value of some loci in the individual string of the population. The next-generation population $P(T+1)$ was obtained after selection, crossover, and mutation.

After the above five steps, the termination condition is judged. If $t=T$, the individual with the greatest fitness obtained in the evolution process is used as the optimal solution output, and the calculation is terminated.

3.3. Power Network Flow Collection and Metadata Generation. In order to establish a scientific power network traffic monitoring model, it is necessary to collect the entire network traffic in real time and quickly judge and process abnormal network traffic, so as to reduce the difficulty and time of power network fault diagnosis and abnormal detection.

According to actual operation requirements, the traffic information is collected through network probes, as shown in Figure 2. According to the hierarchical characteristics of the power network, this model needs to deploy traffic collection probes on the links between the edge networks of different levels of companies and the IDC network, so as to achieve the purpose of obtaining all network traffic comprehensively and accurately.

Distributed hardware probes are deployed on different network nodes and are responsible for the traffic collection of the target network. The flow detection remote management center provides unified network monitoring and network analysis and management. The local visual management platform can perform real-time monitoring and retrospective analysis on the specified network link and can also perform playback analysis on local data packets.

In order to fully obtain the power network flow characteristic data, this paper extracts the flow characteristic metadata from the four dimensions of area, time, business, and link, so that a complete network flow analysis plan can be made. Some of the metadata definitions are shown in Table 2.

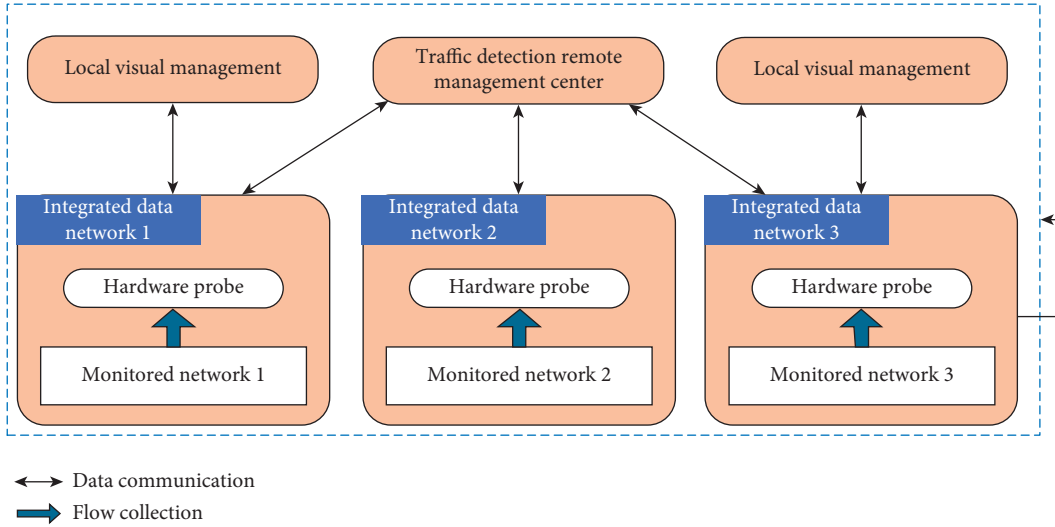


FIGURE 2: Network traffic collection architecture.

4. Method Principle and Process

This section first introduces the process of power network flow metadata preprocessing, then performs statistical analysis on the metadata, and finally focuses on the detailed description of how the GA algorithm optimizes the CNN parameters. The proposed GACNN power network traffic detection method is mainly divided into three stages: data preprocessing, model training, and model verification, and its overall framework is shown in Figure 3. In the preprocessing stage, a series of processing is performed on the collected traffic metadata, including data encoding, data normalization, data shaping, and data splitting. The pre-processed data will be converted into pictures for training and learning in the GACNN model.

4.1. Data Preprocessing. Network metadata are string data composed of multiple information elements extracted by the probe from the original network traffic. Each information element in the metadata occupies a fixed position in the character string, and the character strings are separated by a ^ sign, and the final character string also ends with a ^. For the information element that does not have a value, the position does not need to be filled with any content; that is to say, the two ^ are adjacent at this time.

For example, an extracted piece of network traffic metadata is shown in Figure 4.

For certain feature data of metadata that only contain a few types of data, tag coding can be used to solve it, such as protocol field: http code is 1, ftp code is 2, etc. For some very long strings contained in metadata, for example, the content of the DPI package is “Welcome to the Changsha city!” after byte encoding, it is “235619648990998464306777891175220 0940810994543019102548539974727530785.”

Data standardization can improve the accuracy of the model. At present, there are many data standardization methods. In summary, they can be divided into linear methods (such as extreme value method and standard

deviation method), broken line method (such as three-fold line method), and curve method (such as seminormal distribution). This article mainly adopts the min-max method, and its formula is as follows:

$$y_i = \frac{x_i - \min_{1 \leq j \leq n} \{x_j\}}{\max_{1 \leq j \leq n} \{x_j\} - \min_{1 \leq j \leq n} \{x_j\}}, \quad (8)$$

where x_i is the input data sequence, y_i is the output sequence after the change, and the value is between $[0, 1]$.

For the input of the CNN model, the format should generally be three-dimensional data (height, width, and channel). The data after data encoding and standardization are only some two-dimensional feature vector data. These data cannot be directly used for model training and testing. In this paper, each line of traffic metadata is reshaped according to the proportion of $(50 * 50 * 3)$. When the length of metadata is less than 2500, the same value of metadata is used to fill. The purpose of this is to further increase the reliability of feature extraction. In addition, the channel value is filled according to the data type. For example, if the metadata are an abnormal type in the second category, it is filled with 1, and the normal type is filled with 0. In the multicategory, it is filled with the corresponding data. Figure 5 shows a data image obtained by reshape of a certain piece of traffic metadata.

4.2. Statistical Analysis of Traffic Metadata. In order to quickly and accurately understand the statistical characteristics of the traffic metadata collected by different probes, in addition, the metadata contain 217 explicit characteristics, but they lack some spatiotemporal implicit statistical characteristics, which is not convenient for deep understanding and analysis of abnormal traffic information. In this paper, the characteristics of traffic metadata in different time periods are statistically analyzed, so as to obtain some important supplementary implicit features.

TABLE 2: Some of the metadata definitions of network traffic.

ID	Name	Type	Length	Description	Dimension
1	EventID	String	64	Event unique ID	
2	OccurTime	Long	8	The time when the event first occurred	Time
3	RecentTime	Long	8	Time of the last occurrence of the event	Time
4	SrcGeographyLocationCountryOrRegion	String	128	Country or region of source IP	Area
5	DestGeographyLocationCountryOrRegion	String	128	Country or region of destination IP	Area
6	OriEventType	String	1024	Original event type	Business
7	EventSubType	String	128	Event subtype	Business
8	AbnTrfBaseline	String	20480	Baseline value of abnormal flow	Link
9	ESN	String	64	Link device serial number	Link

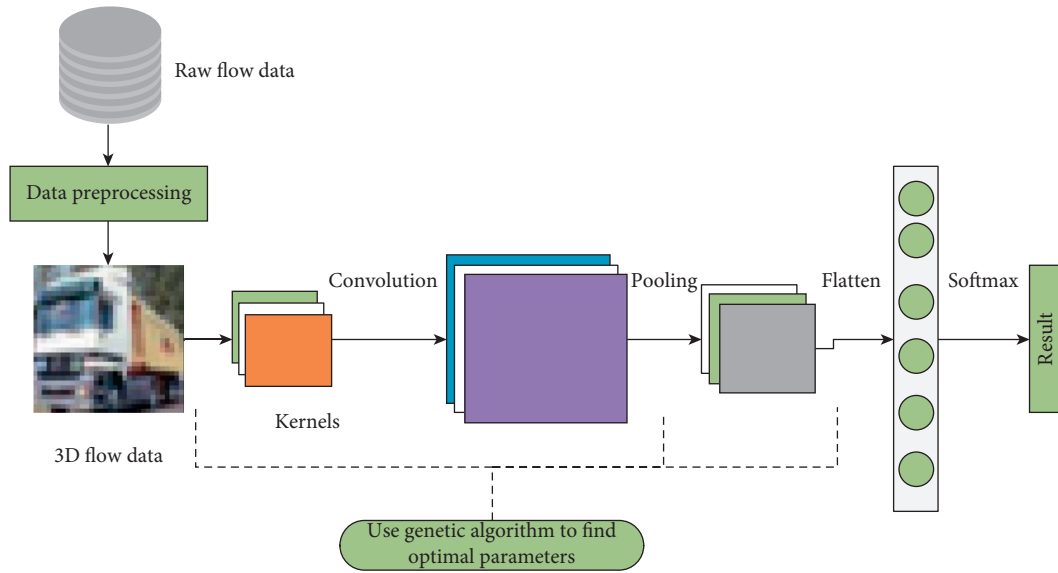


FIGURE 3: The overall framework of GACNN.

```

6^b68c5d479bd7212702000004^10.130.19.183^10.150.24.1^60015^7003^6^1564974
039^1564974041^HTTP^HTTP^GET^http://10.150.24.1:7003/hryw/
assets/mars/dialog/Window.js?version=1.0beta&md5=ba9cfc48f3619eedf4d56899715d748
b^Window.js^5290^ba9cfc48f3619eedf4d56899715d748b^1
564974042643^1^2019-08-05T03:00:42.643^TCP^
10.130.19.183^10.150.24.1^
    
```

FIGURE 4: A piece of element probe traffic metadata.

Figure 6(a) shows the number of uplink bytes of a stream (octetDeltaCount) in the 2019 year, and Figure 6(b) shows the relationship between the number of uplink bytes (octetDeltaCount) and TCP and UDP source ports (sourceTransportPort).

Figure 7 shows the change of traffic ports from July 26, 2019, to August 5, 2019. It can be observed that the flow port is relatively stable, but the fluctuation is relatively large on August 2, 2019.

Figure 8 shows the main distribution of network traffic source ports in 2019 (in Figure 8(a)) and the distribution of traffic source ports in July and August of 2019 (in Figure 8(b)). It can be seen from the right figure that the

source ports are still quite different in July and August of 2019.

Figure 9 shows the statistical distribution of source ports of network traffic, mainly including raw data, rolling average, and rolling standard deviation. It can be seen from the figure that these values are mainly concentrated between 10000 and 25000.

4.3. GA Optimizes the Parameters of CNN. GA is an algorithm that simulates biological evolution for individual selection, crossover, and mutation. Its main core is parameter coding, initial group setting, and fitness function

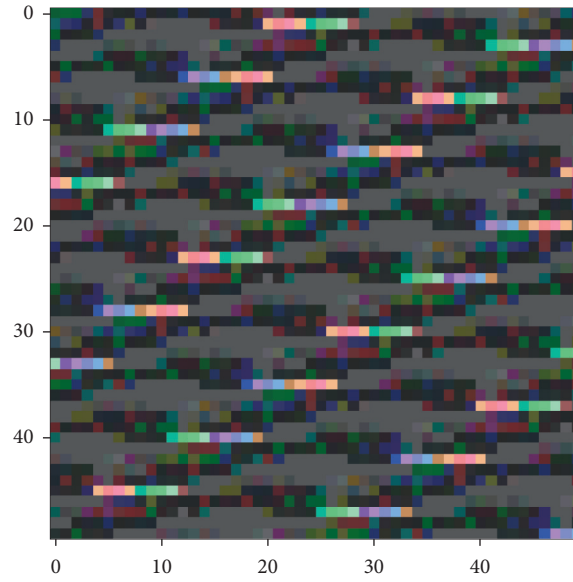


FIGURE 5: A certain flow metadata image after reshape.

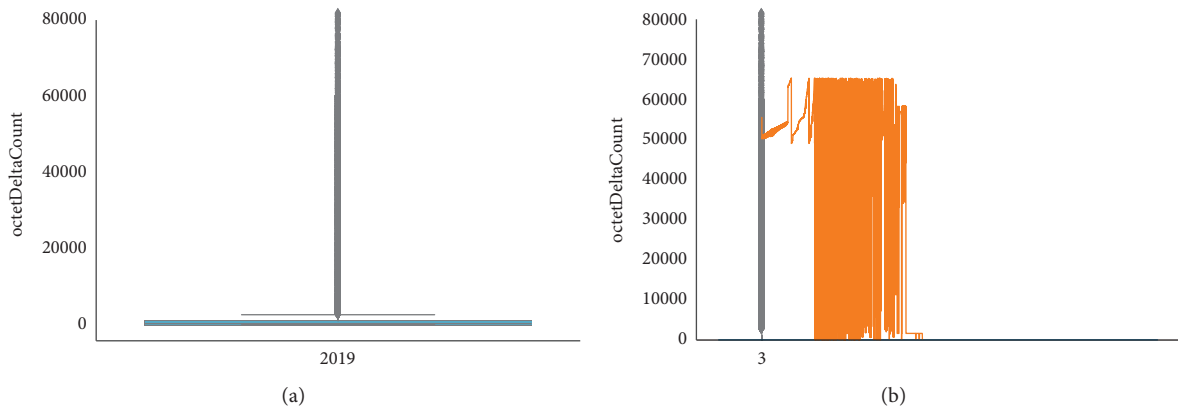


FIGURE 6: The number of uplink bytes of a stream (octetDeltaCount) in 2019. (a) Yearly load. (b) sourceTransportPort.

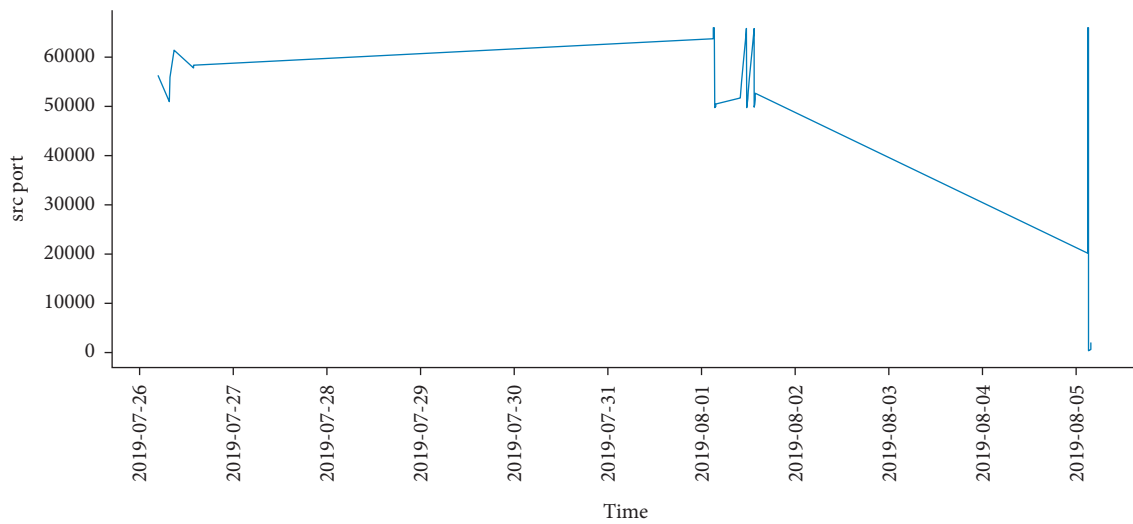


FIGURE 7: The change of traffic ports from July 26, 2019, to August 5, 2019.

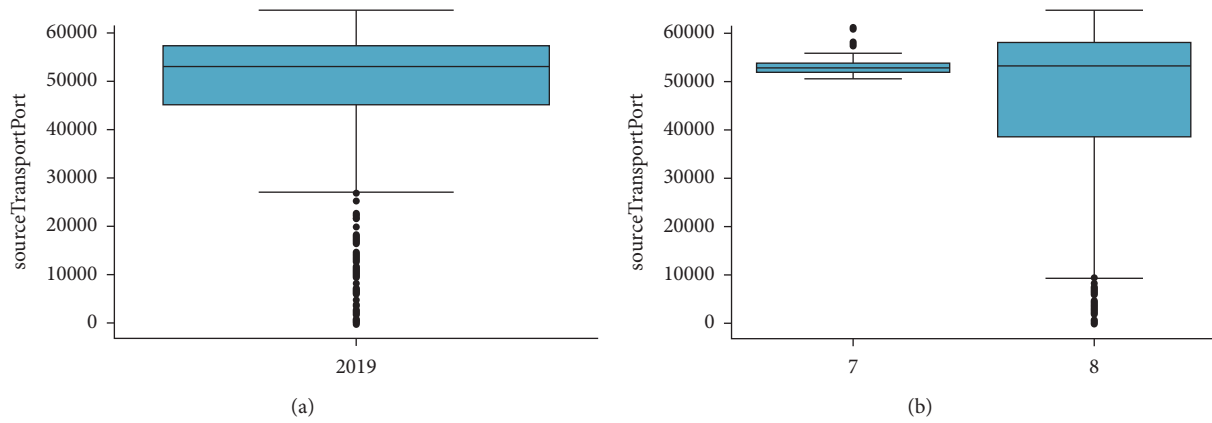


FIGURE 8: The distribution of network traffic source ports in 2019. (a) Yearly load. (b) Month load.

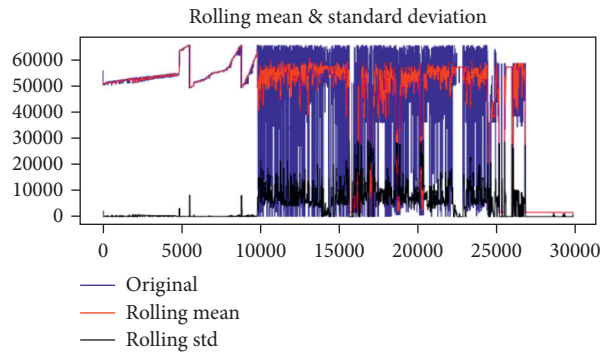


FIGURE 9: The statistical distribution of source ports of network traffic.

determination, and the optimal solution is obtained through the final search. In this paper, the GA is used to optimize the parameters of CNN such as weight, bias, and optimizer selection, and the optimization process is shown in Figure 10.

In the initial population stage, some important parameters of CNN are randomly set, mainly including the filters, the kernel_size, the activation parameters of the Conv2D layer, the loss rate of the dropout layer, and the unit parameters of the dense layer.

- (i) Filters: the number of CNN filters (16, 32, 48, 64).
- (ii) Kernel_size: the shape and size of the filter in CNN. The initial random setting is as follows: (2 * 2), (3 * 3) or (5 * 5).
- (iii) Activation: the activation function used in the CNN model is randomly selected from {relu, selu, elu, sigmoid, tanh}.
- (iv) Loss rate: randomly generated values between [0.1, 0.5].
- (v) Optimizer: select randomly from ["adamax," "adadelta," "adam," "adagrad," "nadam"].
- (vi) Pooling: pooling (none, maximum pooling, average pooling).

- (vii) Loss: the loss is used to match the function of the network, select randomly from ["categorical_crossentropy," "mse," "focal_loss"].

In the selection stage, an individual is randomly selected from the generated population for model training.

In the crossover stage, two crossover points are randomly set in the individual coding string, and then partial gene exchange is performed.

In the mutation stage, the mutation operation is performed on the individual coding string with mutation probability and the value of a random bit.

5. Experiment and Results

In order to evaluate the proposed abnormal traffic detection scheme, this article uses Python, Scikit-learn, NumPy, Pandas, TensorFlow, and Keras to conduct training and testing on a 64 bit Windows computer, which is configured as Intel(R) Core(TM) i3- 4005U 1.7G CPU, 8 GB RAM, 250G solid state drive.

The labeled dataset is a key factor to ensure the performance of deep learning. In order to verify the performance of the GACNN method, the internationally public network

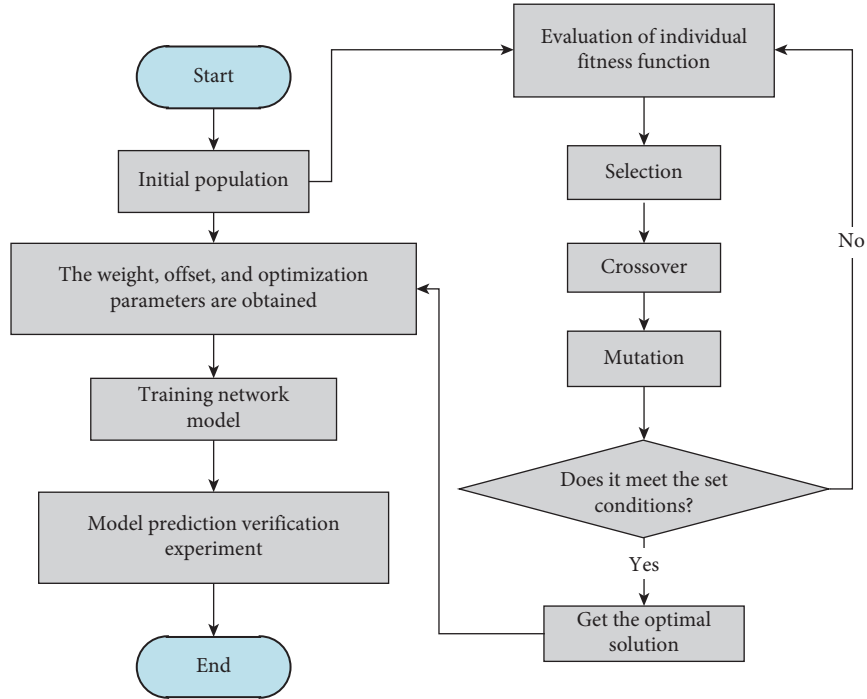


FIGURE 10: The process of optimizing CNN parameters by GA.

intrusion dataset UNSW_NB15 [47] and a part of the traffic dataset are collected by a probe of a Chinese power company.

This article uses accuracy, precision, recall, and F1-scores to evaluate model detection performance. In order to measure the values of these 4 indicators, it is generally calculated by using true positive (TP), false positive (FP), true negative (TN), and false negative (FN).

Accuracy is the proportion of correctly classified samples to the total number of samples, which is defined as

$$\text{Acc} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{TN} + \text{FN}}. \quad (9)$$

Precision: its meaning is the ratio of samples that are actually positive samples among all the samples predicted to be positive, and its expression is

$$P = \frac{\text{TP}}{\text{TP} + \text{FP}}. \quad (10)$$

Recall rate: its meaning is the ratio of positive cases predicted to be positive in a sample that is actually positive, and its expression is

$$R = \frac{\text{TP}}{\text{TP} + \text{FN}}. \quad (11)$$

F1-score: the F1-score considers both the precision rate and the recall rate, so that the two can reach the highest at the same time and strike a balance. The F1-score expression is

$$\text{F1} = \frac{2 \times P \times R}{P + R}. \quad (12)$$

In these formulas, TP is the number of successful detections of the current network traffic category, TN is the number of other network traffic types successfully detected, FP is the number of other network traffic categories identified as the current network traffic category, and FN is the number of current network traffic categories identified as other network traffic categories.

5.1. Tested on UNSW-NB15 Dataset. The original network packets (Pcap files) of the UNSW-NB15 dataset were created by the IXIA PerfectStorm tool in the network-wide laboratory of the UNSW Canberra Network Center, which is used to generate network traffic test of real normal activity and synthetic attack activity. The source files of the dataset are divided based on the simulation dates of January 22, 2015, and February 17, 2015, respectively [47].

The total training dataset selected in the experiment contains 175341 records, and the test set contains 82332 records. These records come from different types of attacks and normal conditions. Furthermore, there are 9 types of the attacks including Analysis, Backdoors, DoS, Exploits, Generic, Reconnaissance, Shellcode, and Worms. All the features in the dataset can be divided into 6 subtypes, namely, flow features, basic features, content features, time features, additional generated features, and labelled features. Table 3 lists the 12 features included in the dataset.

5.1.1. Binary-Classification Test. In order to improve the understanding and analysis of datasets, this paper carries out a series of preprocessing operations before binary

TABLE 3: The 12 features included in the UNSW-NB15 dataset.

No.	Feature name	Type	Feature description	Feature classification
1	srcip	Nominal	Source IP address	Flow features
2	sport	Integer	Source port number	Flow features
3	proto	Nominal	Transaction protocol	Flow features
4	sbytes	Integer	Source to destination bytes	Basic features
5	dbytes	Integer	Destination to source bytes	Basic features
6	swin	Integer	Source TCP window advertisement	Content features
7	smeansz	Integer	Mean of the flow packet size transmitted by the src	Content features
8	sjit	Float	Source jitter (mSec)	Time features
9	djit	Float	Destination jitter (mSec)	Time features
10	ct_flw_http_mthd	Integer	No. of flows that have methods such as get and post in http service.	Additional generated features
11	ct_ftp_cmd	Integer	No of flows that have a command in ftp session	Additional generated features
12	attack_cat	Nominal	The name of each attack category	Labelled features

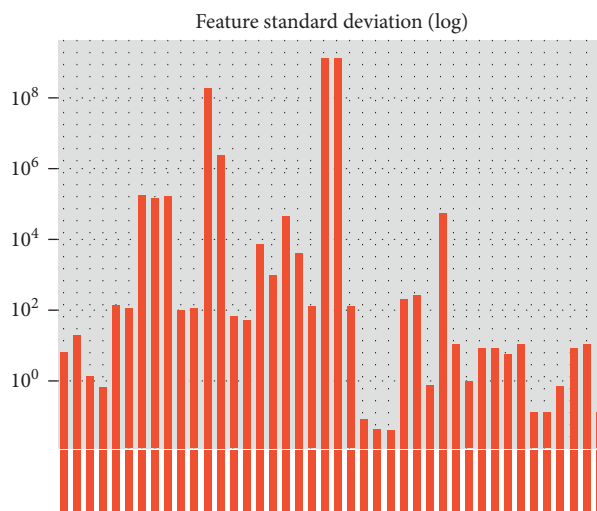


FIGURE 11: The standard deviation of features included in the UNSW-NB15 dataset.

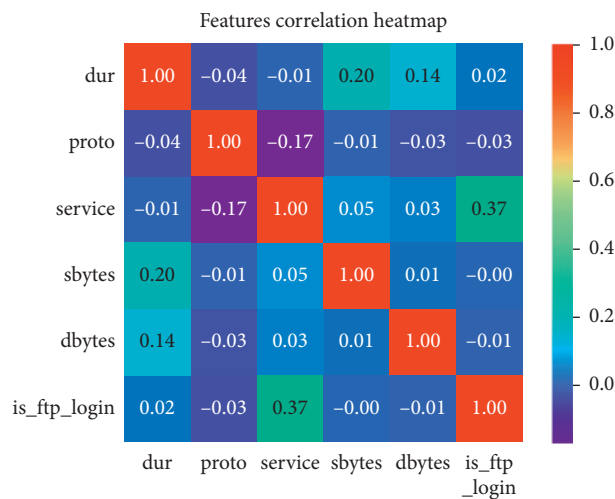


FIGURE 12: Heat map of some features in the UNSW-NB15 dataset.

classification, mainly including standardization, deduplication, dimension reduction, and feature correlation analysis. Among them, the results of standardization and

correlation analysis are shown in Figures 11 and 12. As can be seen from Figure 11, the values of sload, dload, stcpd, dtcpd, and other characteristics after the standard are more

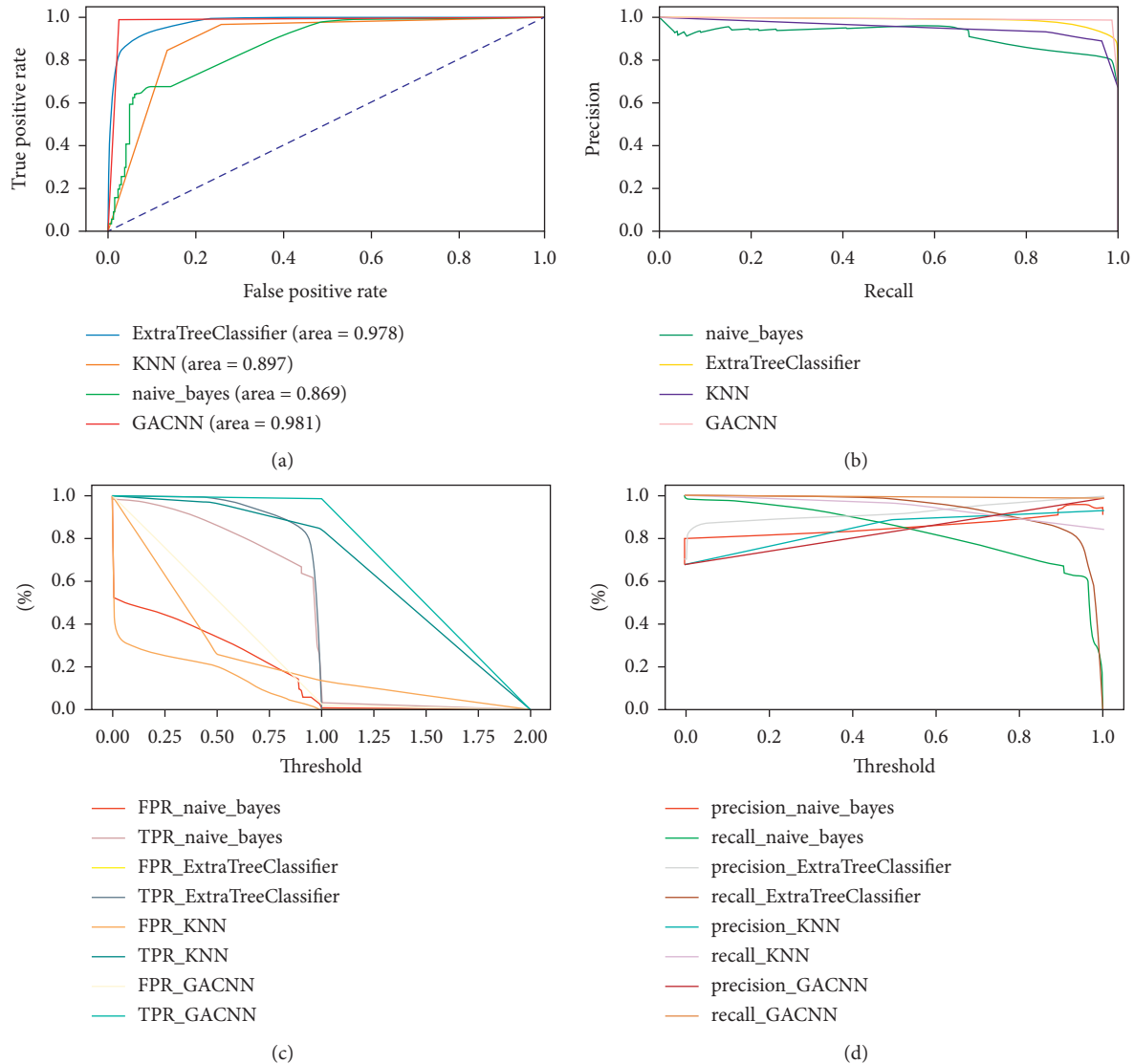


FIGURE 13: Binary-classification results in the UNSW-NB15 dataset.

prominent, reflecting the importance of these characteristics. It can be seen from Figure 12 that the selected eigenvalues have very little correlation. For example, the correlation between proto and service is -0.17 , indicating that the selected eigenvalues are very suitable.

In order to better verify the classification effect of the model, a simple CNN model and some traditional machine learning methods such as ExtraTreeClassifier [48], KNN [49], and naive_bayes [50] are selected for comparison. The experiment uses a dichotomy method from the training dataset to divide the data into two, and both the training and test sets contain (87670, 43) feature data. This experiment is only for the identification of normal and abnormal network flows, and the experimental results are shown in Figure 13.

This experiment is only for the identification of normal and abnormal network flows. This paper selects ROC and AUC curves commonly selected in machine learning to

describe the classification accuracy of the model. The full name of ROC is "Receiver Operating Characteristic." The area of the ROC curve is AUC (Area Under the Curve). AUC is used to measure the performance (generalization ability) of machine learning algorithms for "two classification problems." The most ideal classifier is to classify the sample completely correctly, that is, $FP = 0$ and $FN = 0$. So, the ideal classifier $TPR = 1$ and $FPR = 0$.

The experimental results are shown in Figure 13. It can be seen from the ROC graph in Figure 13 that the AUC value of ExtraTree is 0.978, KNN is 0.897, naive_bayes is 0.869, and GACNN is 0.981. The test results show that the GACNN method is better than the previous three methods.

The same situation also appeared in the test results of the other three indicators. In the graph composed of precision and recall, GACNN is relatively stable when the recall value changes from 0.0 to 1.0, and most of them are fixed at 1.0. However, the other three algorithms fluctuate greatly. In the graph composed

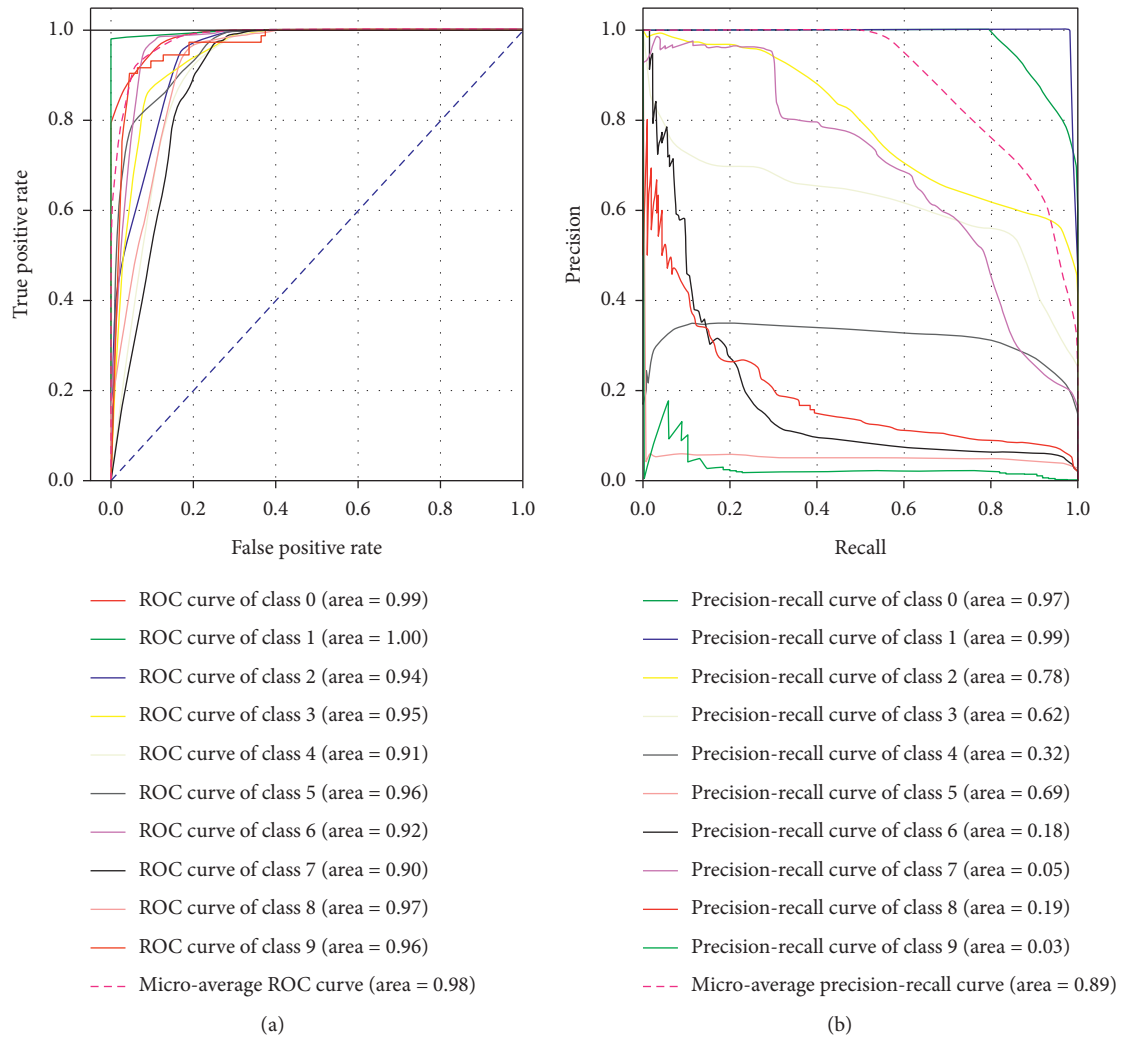


FIGURE 14: The classification results of GACNN.

of threshold and %, when the training threshold is from 0.0 to 1.0, the TPR value of GACNN is basically stable at 1.0. The other three methods are generally lower than 1.0.

5.1.2. Multiclassification Test. In order to accurately classify different network traffic, this paper further tests this dataset and compares the classification effects of the previous four methods in multiclassification. The experiment first performed a series of preprocessing on the metadata and divided the dataset into two subdatasets for training and testing. In addition, both data subsets contain (87670, 43) dimensional feature values and 10 types of attack data. The coding of the 10 attack types is as follows: {0: “Normal,” 1: “Generic,” 2: “Exploits,” 3: “Fuzzers,” 4: “DoS,” 5: “Reconnaissance,” 6: “Analysis,” 7: “Backdoor,” 8: “Shellcode,” and 9: “Worms”}.

The GACNN model has been trained many times to find the optimal parameters of the model and then use the

optimal parameter training and test data. The experimental results of the four methods are shown in Figures 14–17.

In Figure 14, the classification AUC value of the GACNN method for each category exceeds 0.91, and the average AUC value of all types reaches 0.98. In addition, from the precision-recall diagram in Figure 14, it can be seen that AUC values of model classification for 0, 1, 2, 3, and 5 types of network traffic are all more than 0.6, and the average AUC value is 0.89, but the AUC value of 4, 6, 7, 8, and 9 types of network traffic is lower.

In the left subgraph of Figure 15, the classification AUC value of the ExtraTree method for each category is relatively good, basically exceeding 0.6, and the average AUC value of all types also reaches 0.92. In addition, in the right subgraph of Figure 15, the AUC values of the model classification for 0, 1, 2, and 5 types of network traffic are all greater than 0.6, and the average AUC value is 0.74, but the AUC value of 3, 4, 6, 7, 8, and 9 types of network traffic is also lower.

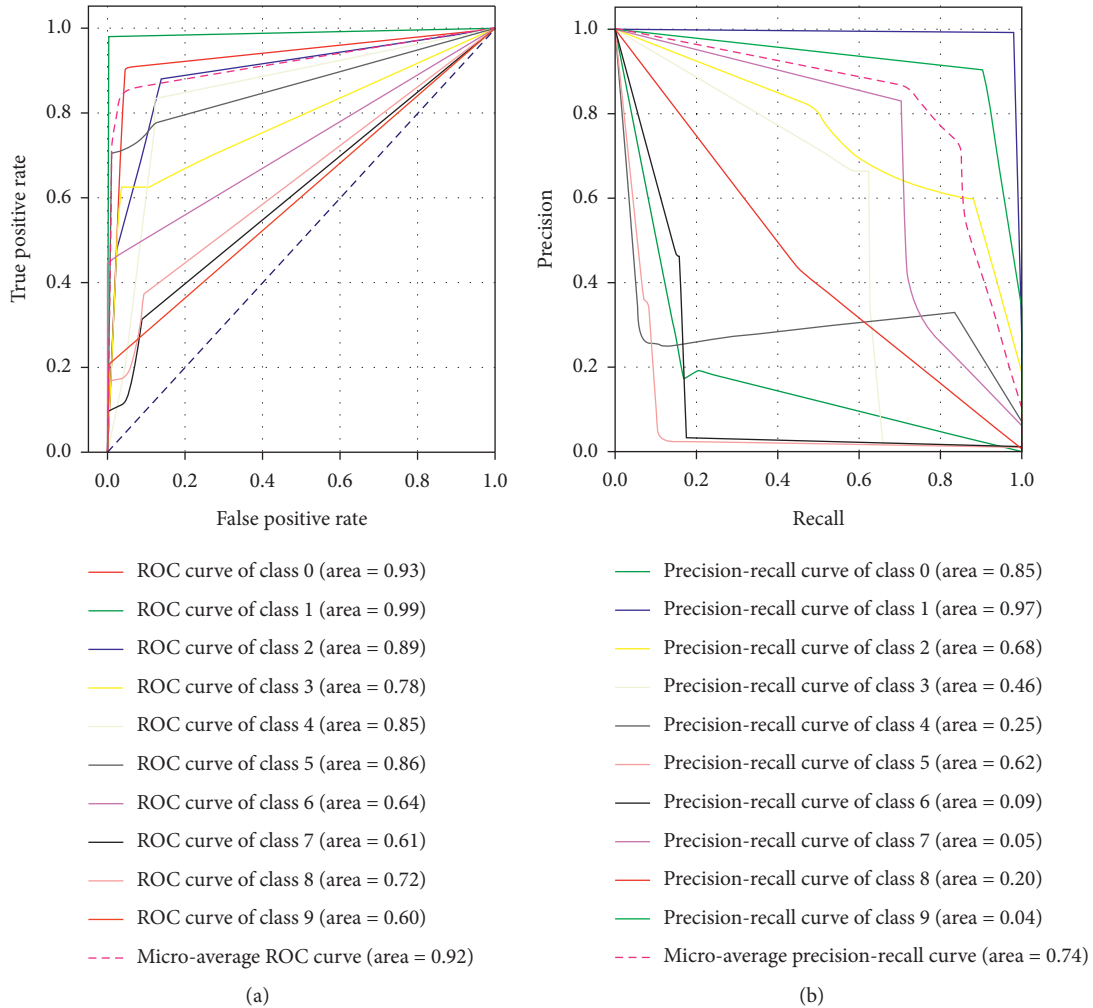


FIGURE 15: The classification results of ExtraTreeClassifier.

In the left subgraph of Figure 16, the classification AUC value of the KNeighbors method for each category is relatively good, basically exceeding 0.6, and the average AUC value of all types also reaches 0.96. In addition, in the right subgraph of Figure 16, the AUC values of the model classification for 0, 1, 2, and 5 types of network traffic are all greater than 0.6, and the average AUC value is 0.86, but the AUC value of 3, 4, 6, 7, 8, and 9 types of network traffic is also lower.

In the left subgraph of Figure 17, the classification AUC value of the naïve_bayes method for each category is relatively balanced, basically exceeding 0.80, but the average AUC value of all types is only 0.82. In addition, in the right subgraph of Figure 15, only the AUC value of the model classification of type 0 network traffic is greater than 0.6, and the AUC values of the other types of network traffic are very low.

From the above experimental results, it can be known that in the multiclassification test of the UNSW-NB15 dataset, the GACNN method is basically better than the other three methods.

5.2. Test on Real Power Dataset. In order to further verify the anomaly classification effect of the proposed method in real network scenarios, the continuous flow data grouping of a power company information network in July and August 2019 was obtained through flow probes and processed and analyzed in the experimental platform. The preprocessed data are used as the input of the experimental platform, and the performance of the algorithm is evaluated through the processing of different algorithms. Some preprocessing results are shown in Figures 5–9.

Due to the different parameters of CNN, the detection performance of the network will be different. In order to evaluate the classification performance of the GACNN method, some initial parameter settings are shown in this experiment in Tables 4 and 5.

The experiment collected normal network flows and three abnormal network flows (C&C attack, SSH brute force attack, and webshell attack). Among them, the normal stream is 563 MB, the webshell stream is 8.8 MB, and the

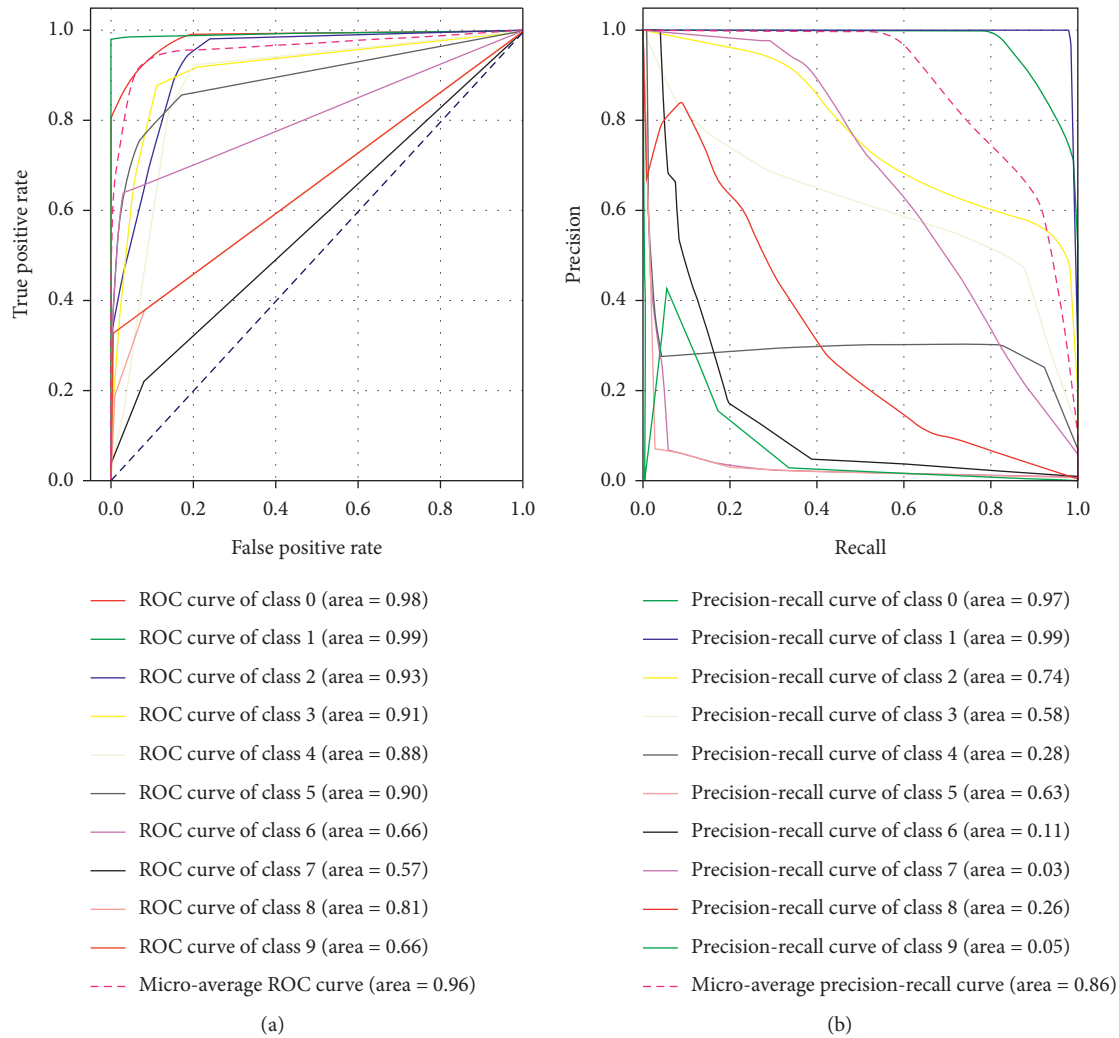


FIGURE 16: The classification results of KNeighborsClassifier.

SSH stream is 5.74 MB. However, C&C is relatively small, only 24 kB. These probe stream metadata are converted into CSV files after preprocessing. In the training and detection stage of the CNN model, the original CSV format data cannot be directly used. Then, merge the CSV files of different attack types and convert them into an NPY file.

In GACNN model training, the dataset is divided into training set and test set, and the training set contains (216066, 50, 50, and 3) records, and the test set contains (72022, 50, 50, and 3) records. The GACNN model contains 5 two-dimensional convolutional layers Conv2D, 5 batch_normalization layers, 5 max_pooling2d layers, a flatten layer, 2 dense layers, and 2 dropout layers.

Like the previous experiment, this experiment also selects the former three machine learning methods for comparison. Table 6 lists the detection results of these

algorithms for three attack types (ExtraTree-E, KNeighbors-K, naïve_bayes-N, and GACNN-G). As can be seen from the results in Table 6, compared with the other three detection methods, the convolution neural network optimization method based on the genetic algorithm has greatly improved the detection rate and unknown attack detection rate, reduced the false alarm rate, and achieved good results.

For example, in binary detection, the precision, recall, and F1-score of GACNN are all higher than 0.96, while other methods are lower than 0.95.

In the multiclassification detection, the precision values of the other three methods are lower than 0.93, recall values are lower than 0.96, and the F1-score values are basically lower than 0.94. On the contrary, the three detection values obtained by GACNN method are relatively good, and all of them reach the values above 0.96.

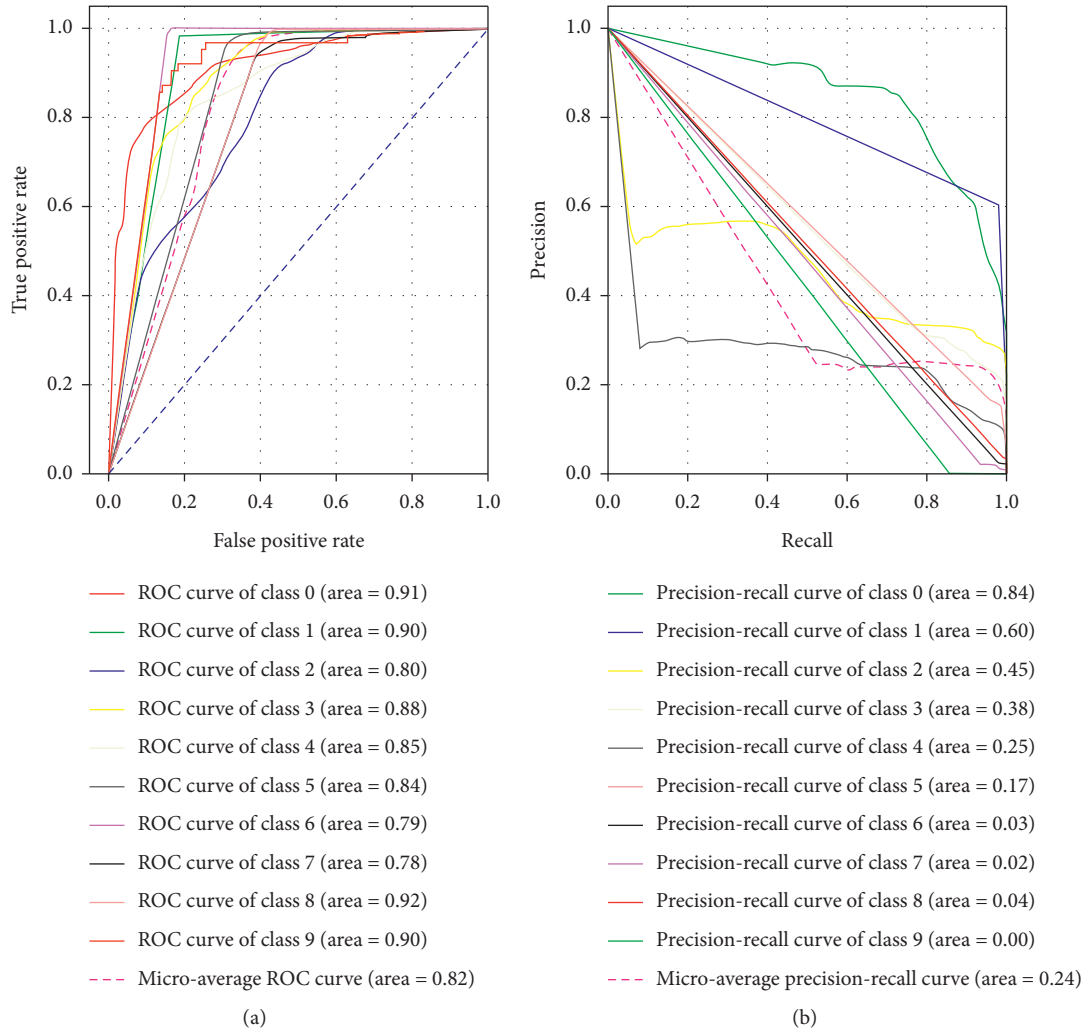


FIGURE 17: The classification results of naive_bayes.

TABLE 4: The initial parameters of CNN.

Parameters	Value
Filters	32, 64
Kernel_size	(3, 3), (5, 5),
Activation	Relu, selu, elu
Input_shape	(50, 50, 3)
Output_shape	2, 10
Dropout_rate	0.1, 0.2, or random number between 0.1 and 0.5
Optimizer	“adamax,” “adadelata,” “adam,” “adagrad”

TABLE 5: The initial parameters of GA.

Parameters	Value
Mutation probability	0.01 or the random number is greater than the threshold value
Crossover probability	0.5
Reproduction algebra	5
Population size	10

TABLE 6: The results of four methods in power network flows.

Class	Type	Precision				Recall				F1-score			
		E	K	N	G	E	K	N	G	E	K	N	G
Two	Normal	0.928	0.921	0.892	1.00	0.962	0.976	0.949	1.00	0.945	0.947	0.920	1.00
	Abnormal	0.917	0.928	0.873	1.00	0.885	0.875	0.837	0.962	0.901	0.901	0.854	0.975
Multi	C&C	0.905	0.805	0.776	0.973	0.911	0.940	0.932	0.993	0.908	0.867	0.846	0.983
	SSH	0.863	0.922	0.892	0.963	0.917	0.905	0.852	0.982	0.889	0.913	0.872	0.972
	Webshell	0.799	0.852	0.925	0.974	0.957	0.950	0.961	0.989	0.870	0.898	0.943	0.981
	Whole	0.847	0.907	0.853	0.978	0.956	0.959	0.950	0.986	0.898	0.932	0.899	0.982

6. Conclusion

With the continuous development of smart grids, the current power information network is constantly expanding, and the possibility of failures in the network is also increasing. Aiming at the high false alarm rate and low detection efficiency of current network traffic anomaly detection, this paper proposes a genetic algorithm-optimized convolutional neural network method to deal with power network traffic anomaly detection.

Compared with the traditional machine learning method, this paper mainly innovates and improves from the following aspects:

- (1) According to the business characteristics and requirements of the power system, this paper establishes a network flow metadata collection model for the power system. This model mainly monitors network traffic and equipment indicator status from the four dimensions of time, area, event, and link, thereby more effectively improving the quality of network services.
- (2) Aiming at the problem that the classification accuracy of CNN depends on parameter settings, this paper proposes to use the genetic algorithm to find the best CNN parameters, which can quickly improve the training accuracy of the CNN method. From the experimental results, this method is superior to other detection methods in the anomaly detection of network flow.
- (3) The classification accuracy of traditional machine learning largely depends on the reasonable selection of network features. The method proposed in this paper uses raw traffic or metadata directly as model input and trains features through self-learning without manual intervention. In addition, the hidden spatiotemporal features and package content analysis can be completed through multiple convolutional learning and spatiotemporal analysis of the model, thereby reducing the complexity of the task and improving the accuracy of the model classification.

In order to further improve the accuracy and efficiency of the method, the next step is to continue to work on the following aspects:

- (1) Improved CNN structure: the network in this algorithm is based on a simple 2-dimensional CNN model. In future work, you can try to use ResNet, VGGNet, and GoogLeNet models to build deeper mixed networks to further improve classification accuracy.

- (2) Continue to increase the testing of the method in large-scale power network traffic to realize practical application in engineering.

Data Availability

Two datasets are used in the paper, among which UNSW-NB15 can be downloaded directly from the Internet and the other is the enterprise internal test dataset. The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the key scientific and technological project of "Research and Application of Key Technologies for Network Security Situational Awareness of Electric Power Monitoring System of China Southern Power Grid Corporation, (no. ZDKJXM20170002)" the teaching reform and scientific research project of "OBE-Based Engineering Education Professional Certification and Evaluation Information System" of Changsha University of Science and Technology, and Open fund project of the Hunan Provincial Engineering Research Center of Electric Transportation and Smart Distribution Network (Changsha University of Science and Technology) (no. 3040102-1105008).

References

- [1] J. Liu and J. Weng, "Review of smart grid security," *Information Network Security*, vol. 5, pp. 78–84, 2016.
- [2] X. Wang and X. Cheng, "Overview of network security technology in smart grid," *Computer CD Software and Application*, vol. 20, pp. 159–161, 2013.
- [3] W. Wang and Z. Lu, "Cyber security in the smart grid: survey and challenges," *Computer Networks*, vol. 57, no. 5, pp. 1344–1371, 2013.
- [4] L. Xu, "Network communication architecture and key technologies of smart grid," *Electrical Technology*, vol. 8, pp. 16–20, 2010.
- [5] Z. Xia, J. Tan, K. Gu, and W. Jia, "Detection resource allocation scheme for two-layer cooperative IDSs in smart grids," *Journal of Parallel and Distributed Computing*, vol. 147, pp. 236–247, 2021.

- [6] M. A. Faisal, Z. Aung, J. R. Williams, and A. Sanchez, "Securing advanced metering infrastructure using intrusion detection system with data stream mining," *Lecture Notes in Computer Science*, Springer, vol. 7299 Berlin, Germany, , 2012.
- [7] H. Kasai, W. Kellerer, and M. Kleinsteuber, "Network volume Anomaly detection and identification in large-scale networks based on online time-structured traffic tensor tracking," *IEEE Transactions on Network and Service Management*, vol. 13, no. 3, pp. 636–650, 2016.
- [8] L. Wu, S. Zhang, T. Lei, and Y. Cai, "A traffic detection model for power system data network," *Information and Communication*, vol. 7, pp. 45–46, 2013.
- [9] Y. Zhang, X. Li, D. Li et al., "Abnormal flow detection of industrial control network based on convolutional neural network," *Computer Application*, vol. 39, no. 5, pp. 1512–1517, 2019.
- [10] D. Li, X. Wang, B. Yu, and T. Huang, "Network traffic classification method based on one-dimensional convolution neural network," *Computer Engineering and Applications*, vol. 56, no. 3, pp. 94–99, 2020.
- [11] P. Wang, Q. Zheng, G. Niu et al., "Port scan detection algorithm based on traffic statistics," *Acta Communication Sinica*, vol. 28, no. 12, pp. 14–18, 2007.
- [12] Z. Sun, J. Zhai, and Y. Dai, "An encryption flow identification method based on DPI and load randomness," *Journal of Applied Sciences*, vol. 37, no. 5, pp. 711–720, 2019.
- [13] J. Garcia, "A clustering-based analysis of DPI-labeled video flow characteristics in cellular networks," in *Proceedings of the 2017 IFIP/IEEE Symposium on Integrated Network and Service Management (IM)*, pp. 991–994, Lisbon, Portugal, May 2017.
- [14] Y. Wang and X. Wei, "A security model of ubiquitous power internet of things based on SDN and DFI," in *Proceedings of the 2020 Information Communication Technologies Conference (ICTC)*, pp. 55–58, Nanjing, China, May 2020.
- [15] H. Chen, Z. Hu, Z. Ye, and W. Liu, "A new model for P2P traffic identification based on DPI and DFI," in *Proceedings of the 2009 International Conference on Information Engineering and Computer Science*, pp. 1–3, Wuhan, China, December 2009.
- [16] X. Liu, Z. Tang, and B. Yang, "Predicting network attacks with CNN by constructing images from NetFlow data," in *Proceedings of the 2019 IEEE 5th Intl Conference on Big Data Security on Cloud (BigDataSecurity), IEEE Intl Conference on High Performance and Smart Computing, (HPSC) and IEEE Intl Conference on Intelligent Data and Security (IDS)*, pp. 61–66, Washington, DC, USA, May 2019.
- [17] C. Xu, Z. Su, Q. Jia, D. Zhang, Y. Xie, and A. Yang, "Neural dialogue model with retrieval attention for personalized response generation," *Computers, Materials & Continua*, vol. 62, no. 1, pp. 113–122, 2020.
- [18] W. Jiang, Y. Wang, Y. Jiang et al., "Mobile internet mobile agent system dynamic trust model for cloud computing," *Computers, Materials & Continua*, vol. 62, no. 1, pp. 123–136, 2020.
- [19] Y. Xu, X. Meng, Y. Li, and X. Xu, "Research on privacy disclosure detection method in social networks based on multi-dimensional deep learning," *Computers, Materials & Continua*, vol. 62, no. 1, pp. 137–155, 2020.
- [20] S. Weng, Y. Liu, Y. Shi, B. Ou, C. Zhang, and C. Wang, "A general framework of reversible data hiding with controlled contrast enhancement," *Computers, Materials & Continua*, vol. 62, no. 1, pp. 157–177, 2020.
- [21] K. Gu, X. Dong, and L. Wang, "Efficient traceable ring signature scheme without pairings," *Advances in Mathematics of Communications*, vol. 14, no. 2, pp. 207–232, 2020.
- [22] T. Wang, D. Zhao, and Y. Feng, "Two-stage multiple kernel learning with multiclass kernel polarization," *Knowledge-Based Systems*, vol. 48, pp. 10–16, 2013.
- [23] T. Wang, L. Zhang, and W. Hu, "Bridging deep and multiple kernel learning: a review," *Information Fusion*, vol. 67, pp. 3–13, 2021.
- [24] F. Yu, S. Qian, X. Chen et al., "Chaos-based engineering applications with a 6D memristive multistable hyperchaotic system and a 2D SF-simm hyperchaotic map," *Complexity*, vol. 2021, Article ID 6683284, 21 pages, 2021.
- [25] N. Liao, Y. Song, S. Su, X. Huang, and H. Ma, "Detection of probe flow anomalies using information entropy and random forest method," *Journal of Intelligent and Fuzzy Systems*, vol. 39, no. 1, pp. 433–447, 2020.
- [26] Z. Tang, X. Zeng, J. Chen, and Z. Guo, "Survey of network traffic analysis based on machine learning," *Network New Media Technology*, vol. 9, no. 5, pp. 1–8, 2020.
- [27] J. Hou, P. Fu, Z. Cao, and A. Xu, "Machine learning based DDoS detection through NetFlow analysis," in *Proceedings of the MILCOM 2018—2018 IEEE Military Communications Conference (MILCOM)*, pp. 1–6, Los Angeles, CA, USA, October 2018.
- [28] K. Flanagan, E. Fallon, A. Awad, and P. Connolly, "Self-configuring NetFlow anomaly detection using cluster density analysis," in *Proceedings of the 2017 19th International Conference on Advanced Communication Technology (ICACT)*, pp. 421–427, PyeongChang, South Korea, February 2017.
- [29] J. Fei, T. Zhang, Y. Ma, and C. Zhou, "A DDoS attack detection method for power grid industrial control system based on BF-DT-CUSUM algorithm," *Telecom Science*, vol. 31, no. s1, pp. 106–112, 2015.
- [30] Y. Xu, "Real time traffic classification method of power business based on improved random forest algorithm," *Power System Protection and Control*, vol. 44, no. 24, pp. 82–89, 2016.
- [31] J. Du, W. Su, and Q. Peng, "Anomaly detection of power integrated data network based on traffic structure," *Electronic Technology and Software Engineering*, vol. 18, no. 49, pp. 1–12, 2014.
- [32] Z. Wu and Y. Dong, "Research on feature selection method of network video traffic classification," *Computer Engineering and Application*, vol. 54, no. 6, pp. 7–13, 2018.
- [33] A. Tamer, P. Dilina, and A. Mark, "An evaluation of the performance of restricted Boltzmann machines as a model for anomaly network intrusion detection," *Computer Networks*, vol. 144, pp. 111–119, 2018.
- [34] X. Sun, S. Ma, Y. Li et al., "Enhanced echo-state restricted Boltzmann machines for network traffic prediction," *IEEE Internet of Things Journal*, vol. 7, no. 2, pp. 1287–1297, 2020.
- [35] Q. P. Nguyen, K. W. Lim, D. M. Divakaran, K. H. Low, and M. C. Chan, "GEE: a gradient-based explainable variational autoencoder for network anomaly detection," in *Proceedings of the 2019 IEEE Conference on Communications and Network Security (CNS)*, pp. 91–99, Washington, DC, USA, June 2019.
- [36] R. Dargenio, S. Srikant, E. Hemberg, and U. O'Reilly, "Exploring the use of autoencoders for botnets traffic representation," in *Proceedings of the 2018 IEEE Security and Privacy Workshops (SPW)*, pp. 57–62, San Francisco, CA, USA, May 2018.
- [37] W. Zhang, Y. Yu, Y. Qi, F. Shu, and Y. Wang, "Short-term traffic flow prediction based on spatio-temporal analysis and

- CNN deep learning,” *Transportmetrica A: Transport Science*, vol. 15, no. 2, pp. 1688–1711, 2019.
- [38] F. Ertam and E. Avcı, “A new approach for internet traffic classification: GA-WK-ELM,” *Measurement*, vol. 95, pp. 135–142, 2017.
- [39] W. Wang, M. Zhu, J. Wang, X. Zeng, and Z. Yang, “End-to-end encrypted traffic classification with one-dimensional convolution neural networks,” in *Proceedings of the 2017 IEEE International Conference on Intelligence and Security Informatics*, Beijing, China, July 2017.
- [40] D. Gao and S. Yao, “Analysis and prediction of business flow of power backbone communication transmission network,” *Automation and Instrumentation*, vol. 12, pp. 214–217, 2017, in Chinese.
- [41] H. Lv, J. Fan, and X. Ma, “Information flow monitoring and prediction analysis platform for electric power communication network,” *Science and Technology Innovation*, vol. 16, pp. 79–80, 2020, in Chinese.
- [42] J. Lin, S. Fan, Z. Xu et al., “Power grid operation situation awareness evaluation model based on fuzzy analytic hierarchy process and LSTM-attention mechanism,” *Electric Power Information and Communication Technology*, vol. 18, no. 4, pp. 58–66, 2020, in Chinese.
- [43] Y. LeCun, K. Kavukcuoglu, and C. Farabet, “Convolutional networks and applications in vision,” in *Proceedings of the 2010 IEEE International Symposium on Circuits and Systems*, pp. 253–256, Paris, France, May 2010.
- [44] C. Wang, Z. Wang, Q. Duan et al., “Photovoltaic power prediction based on convolution long short memory hybrid neural network optimized by genetic algorithm,” *Acta Physiologica Sinica*, vol. 69, no. 10, pp. 143–149, 2020.
- [45] F. Amini and G. Hu, “A two-layer feature selection method using genetic algorithm and elastic net,” *Expert Systems with Applications*, vol. 166, Article ID 114072, 2021.
- [46] N. Maleki, Y. Zeinali, and S. T. A. Niaki, “A k-NN method for lung cancer prognosis with the use of a genetic algorithm for feature selection,” *Expert Systems With Applications*, vol. 164, Article ID 113981, 2021.
- [47] M. Nour and J. Slay, “UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set),” in *Proceedings of the Military Communications and Information Systems Conference (MilCIS)*, November 2015.
- [48] J. Sharma, C. Giri, O. C. Granmo, and M. Goodwin, “Multi-layer intrusion detection system with ExtraTrees feature selection, extreme learning machine ensemble, and softmax aggregation,” *EURASIP Journal on Information Security*, vol. 15, 2019.
- [49] N. Sameera and M. Shashi, “Encoding approach for intrusion detection using PCA and KNN classifier,” *Advances in Intelligent Systems and Computing*, vol. 1090, pp. 187–199, 2020.
- [50] G. Piraisoody, C. Huang, B. Nandy, and N. Seddigh, “Classification of applications in HTTP tunnels,” in *Proceedings of the 2013 IEEE 2nd International Conference on Cloud Networking (CloudNet)*, pp. 67–74, San Francisco, CA, USA, November 2013.

Research Article

Early Rumor Detection Based on Deep Recurrent Q-Learning

Wei Wang , Yuchen Qiu, Shichang Xuan , and Wu Yang 

Information Security Research Center, College of Computer Science and Technology, Harbin Engineering University, Harbin 150001, China

Correspondence should be addressed to Wu Yang; yangwu@hrbeu.edu.cn

Received 1 March 2021; Accepted 19 May 2021; Published 1 June 2021

Academic Editor: Lu Liu

Copyright © 2021 Wei Wang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Online social networks provide convenient conditions for the spread of rumors, and false rumors bring great harm to social life. Rumor dissemination is a process, and effective identification of rumors in the early stage of their appearance will reduce the negative impact of false rumors. This paper proposes a novel early rumor detection (ERD) model based on reinforcement learning. In the rumor detection part, a dual-engine rumor detection model based on deep learning is proposed to realize the differential feature extraction of original tweets and their replies. A double self-attention (DSA) mechanism is proposed, which can eliminate data redundancy in sentences and words at the same time. In the reinforcement learning part, an ERD model based on Deep Recurrent Q-Learning Network (DRQN) is proposed, which uses LSTM to learn the state sequence features, and the optimization strategy of the reward function is to take into account the timeliness and accuracy of rumor detection. Experiments show that, compared with existing methods, the ERD model proposed in this paper has a greater improvement in the timeliness and detection rate of rumor detection.

1. Introduction

With the rapid development of the Internet, social networks and people's lives have become increasingly close, and the participation and utilization rate of netizens has risen rapidly [1]. The global digital statistics report [2] released by "We Are Social" in 2019 shows that, by the end of 2018, there were 3.48 billion social network users in the world, accounting for 45% of the world's total population. Social network platforms represented by Twitter and Weibo provide netizens with functions to post information and express opinions. News media have gradually established official accounts on social networks for news reporting, so social networks have gradually become people's main sources of information.

5G and edge computing bring certain security issues to social networks [3, 4]. From the content point of view, the popularity of social networks improves life efficiency, but it has also become an environment for online rumors. An early study of social psychology defined rumors as "propositions spread without verification by relevant departments" [5]. Today, with the explosion of information, a huge amount of

information is spread on social networks every day, including a lot of rumors. The harm of rumors to society cannot be ignored. For example, the 2011 Japanese earthquake triggered a tsunami that caused the Fukushima nuclear power plant to explode. The incident had little impact on China, but some illegal traders took the opportunity to drive up salt prices on the grounds that sea salt was contaminated by nuclear power. Salt prices in many places across the country have soared, and many people have been incited to rob salt in salt farms, which has seriously disrupted the order of social life. The purpose of the rumors is generally destructive, such as pranks and revenge on society. Because social networks have the characteristics of virtuality and anonymity, the cost of creating and spreading rumors is extremely low, and online rumors have become a trend of flooding. In the face of rumors rampant in the network environment, social network platforms have established rumor-defying accounts to manually refute rumors, but only relying on manual review to stop rumors is not only high in labor costs but also very inefficient, so artificial intelligence-based rumor detection technology has gradually become research hot spot.

The spreading process of rumors has obvious timeliness [6]. Specifically, rumors spread rapidly in the form of outbreaks when they appeared in the early days, but over time, their spread speed will be greatly reduced until they eventually die out. Figure 1 shows the spreading sequence diagram of a Twitter rumor. The red text represents the original tweet, the yellow text represents the reply message that questioned the original tweet, and the green text represents the reply message that opposes the original tweet.

The rumor was released after a shooting incident. The general content of the rumor was “According to the police, there were a large number of shooters in the shooting.” Later, after investigation by relevant departments, there was only one shooter, and the police did not disclose the information. Within two hours after the rumor was released, there were a thousand reposts, and the rumor spread to tens of thousands of people. Analyzing the spreading process of this rumor, when there is obvious opposition and questioning information in the comment area, the information can be preliminarily judged as a rumor. If rumors can be accurately identified and their spreading behavior can be controlled when they appear early, the adverse effects of false rumors can be greatly reduced. Therefore, early rumors detection research on social networks is very important.

The main contributions of this paper are as follows. Aiming at the difference in content characteristics between original tweets and reply messages in Twitter, a dual-engine rumor detection model based on the self-attention mechanism is proposed, which improves the accuracy of rumor detection; in addition, we propose an early rumor detection (ERD) based on recurrent Q-learning, which can detect rumors earlier with higher accuracy.

The remainder of this paper is organized as follows: Section 2 briefly introduces the related works and research issues; Section 3 proposes an ERD model based on deep recurrent Q-learning and a dual-engine rumor detection model based on self-attention mechanism; Section 4 discusses and analyzes the experiment results. Section 5 draws the conclusion and proposes future research directions.

2. Related Works

2.1. Rumor Detection. The essence of the rumor detection problem is text classification. The current research on rumor detection is divided into two categories: methods based on traditional machine learning and methods based on deep learning. The former generally uses methods such as naive Bayes classification, decision trees, and support vector machines. Castillo et al. [8] used machine learning algorithms based on feature engineering to classify rumors and extracted a large number of text features based on the characteristics of rumors, including text length and the number of likes. On the basis of this method, many scholars [9–14] began to try to use different machine learning algorithms and richer features to study rumor detection.

Although the method based on machine learning can solve the problem of rumor detection to a certain extent, it is time-consuming, laborious, and inefficient in the feature engineering stage. The quality of the feature heavily relies on

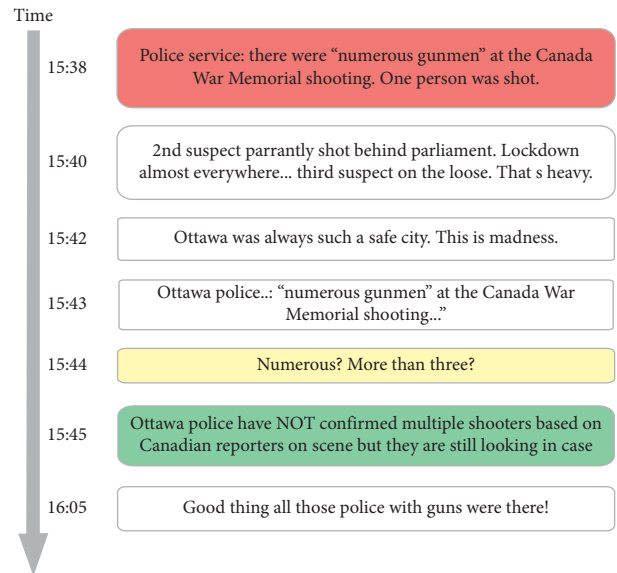


FIGURE 1: A rumor from PHEME dataset [7].

manual experience, which affects the quality of the rumor detection model. With the widespread application of deep learning in the field of natural language processing, researchers have also begun to use deep learning methods to solve the problem of rumor detection. The reply information of tweets has a great influence on the effect of rumor detection. Related researchers have proposed a multitask learning model for rumor detection and user stance detection. The most typical method is the multitask joint learning model proposed by Ma et al. [15], which is to define a shared layer as a bridge between the rumor detection deep learning model and the user stance detection deep learning model to exchange information. Li et al. [16] added user characteristics and attention mechanism on this basis to improve the performance of the model. In addition, with the emergence of the BERT (Bidirectional Encoder Representation from Transformers) language model, the rumor detection methods based on BERT have been proposed, such as the model of Yu et al. [14]. In addition, the rumor detection model based on the propagation tree has gradually attracted the attention of researchers. Its starting point is to abstract the tweets into a propagation tree according to the timeline and convert the rumors classification into tweet tree classification, such as Ma et al. [17] and Kumar’s [18] research work.

2.2. ERD. In the current research related to rumor detection, the research focus is mainly on improving the accuracy of rumor detection, and the early detection of rumors is less involved. The current ERD methods can be divided into three types:

- (1) Real-time rumor detection, such as the model proposed by Castillo et al. [19]. This model uses a support vector machine to classify the original information without considering the reply information, so there is no delay in the detection time, so as

to achieve real-time detection. Although the real-time rumor detection method can ensure the detection of rumors in the early stage, it has a high rate of misjudgment and has little practical value.

- (2) ERD based on static checkpoints. For example, Dungs et al. [20] proposed a detection method based on a hidden Markov model. This method uses a fixed number of replies as an interval when the model reads the reply information (the interval length used in the literature is 5). Set a static checkpoint; each checkpoint will consider whether to output the detection result. If the detection result outputs, the rumor detection process ends; otherwise, the reply information will continue to be read until a detection point appears and the result is output. Although this method can theoretically achieve early detection, it is not flexible enough to give play to the potential performance of the model.
- (3) ERD based on reinforcement learning. For example, the model proposed by Zhou et al. [21] consists of two parts: a rumor detection module (RDM) and a checkpoint module (CKM). The CKM is implemented by a reinforcement learning model, which dynamically controls the number of input replies from the RDM enables ERD. The model can use the reinforcement learning method to constantly weigh the detection time and detection accuracy to achieve the best balance between the “early nature” and accuracy of rumor detection.

2.3. Problems. The original tweet and the reply message are two completely different messages. The original tweet is generally a complete description of an event, whose expression is more rigorous. Reply messages are towards the original tweet, sometimes even an emoticon or a punctuation mark. Existing models generally ignore the difference between the original tweets and their reply information, even though individual multitask joint learning models (such as Ma et al. [13]) use two independent networks to process the original tweet and the reply information, but these two independent networks are often two networks with the same structure. For these two kinds of information with large differences, it is not reasonable to use the same network for modeling, especially for the expression of chaotic word order such as comment information. If the recurrent neural network is only used for modeling, many potential features will be ignored. Therefore, it is necessary to separately model the characteristics of tweets and reply messages.

In terms of ERD, this paper is based on a reinforcement learning strategy to solve the problem of ERD, because the ERD model proposed by Zhou et al. [21] applies reinforcement learning to the ERD problem. The ERD model is composed of CKM and RDM. The CKM is implemented by DQN to control the number of reply messages input to

RDM. The RDM is implemented by GRU. Through in-depth analysis of the model, it is found that the model has the following problems:

- (1) Potential meaning of the state sequences are ignored

Reinforcement learning is generally based on the “Markov decision process,” but in the case of ERD, it is more reasonable to regard the ERD process as a “partially observable Markov decision process” because the state sequence generated by RDM is potentially helpful for ERD.

Specifically, the state sequence refers to the coding sequence of known information formed by continuously inputting reply information to the RDM. For each state in the state sequence, the RDM can output the rumor classification result corresponding to the state. But for different states, even if the classification result output by the RDM is the same, the corresponding probabilities of the results are different. For this different probability sequence, the probability value may show a steady upward trend. For example, when RDM reads 2 reply messages, the probability of RDM outputting rumors and nonrumors is 0.55 and 0.45 (Softmax does the final classification; the sum of the two probabilities is 1), and the classification result is a rumor, but when 4 messages are read, the probability that RDM will output rumors and nonrumors is 0.85 and 0.15, and the classification result is still rumors; the difference is that the model will become more “certain.” For the case where the classification probability changes steadily (the probability of one label increases; the probability of the other label will inevitably decrease because the probability sum is 1), the model can be allowed to output the detection results in advance, which can more effectively avoid the rumor detection process. By observing the partial change trend of the state to represent the actual state of the environment, this process is actually a partial Markov decision process. The CKM in ERD is implemented by DQN based on the Markov decision process, and the sequence features cannot be obtained, resulting in the model’s poor performance in the timeliness of rumor detection.

- (2) Incomplete reward function

The number of rumors in social networking platforms is much less than the number of nonrumors, so rumor detection is actually anomaly detection. In the field of anomaly detection, a model with a higher accuracy rate may be not the best one, and a model with a higher recall rate is often more practical [22]. The reward function used by ERD has the problem of uneven sample distribution, which leads to low model recall. In addition, there is

the problem of insufficient flexibility in the “early” detection strategy. The reward function of ERD is as follows:

$$r_i = \begin{cases} \log M & \text{terminal with correct prediction,} \\ -P & \text{terminal with wrong prediction,} \\ -\varepsilon & \text{continue.} \end{cases} \quad (1)$$

If the decision action is to terminate the reading, there will be two situations. If the prediction is correct, a reward of $\log M$ is obtained, where M is the number of samples whose predictions are correct; if the prediction fails, the penalty is $-P$, where P is a constant 100. If the action is to continue reading, it will be punished by $-\varepsilon$, where ε is 0.01. This function specifically has the following problems:

- (1) Predicting the correct reward of $\log M$ is intended to keep the model in a good state of performance in stages and to make the model converge as soon as possible, but the problem is that the model converges to the local optimal value, which reduces the generalization ability of the model.
- (2) The prediction error is punished by $-P$; the purpose is to make the model make a “more cautious” judgment because once the prediction error is punished, the penalty is great. However, the model also has the problem of poor generalization ability, because when the number of rumors is small, if the misrecognition of rumors and misrecognition of nonrumors are treated equally, the recall rate of the model will decrease.
- (3) Continuing to read the reply information is punished by $-\varepsilon$. The original intention is to make the model output the result as soon as possible, but the problem is that the model does not perceive the “urgency” of time. In other words, for a rumor message, if the model has read 2 replies and 50 replies, the penalty for continuing to read is the same. But in fact, the penalty should be greater after reading 50 replies, because the rumors may have spread over time and the results need to be reached as soon as possible.

3. Proposed Model and Algorithm

3.1. Problem Description. The goal of ERD is to achieve higher accuracy of rumor detection by collecting less information. This problem can be described as follows: set the input tweet as $X = \{x_0, x_1, \dots, x_n\}$, where x_0 represents the original tweet; others represent the reply information related to the original tweet and are sorted in chronological order. x_i is composed of text information and metadata information. The classification result $y = \{\text{rumor,}$

$\text{nonrumor}\}$, and $t \in [0, n]$ represents the number of reply messages used in the rumor detection process, so t indirectly expresses the time-consuming detection process. Therefore, the purpose of ERD can be described as, for input X , it is necessary to accurately output tweet type y when t is as small as possible.

3.2. Early Rumor Detection Model Based on DRQN. In order to effectively solve the problems mentioned in Section 2, we propose an ERD model based on DRQN (Deep Recurrent Q-Learning Network). The basic model architecture is shown in Figure 2, which consists of a RDM and a control model.

3.2.1. Control Model. The control model is implemented by DRQN, which is a typical partially observable Markov decision algorithm. The recurrent neural network enables the model to have the memory function of the state sequence and then can learn the potential features in the state sequence. In the control module, this paper uses LSTM to realize the memory function of the state sequence. LSTM obtains the actions it considers reasonable by observing the state information and the last judgment.

The specific calculation process is shown in the following formula:

$$\begin{aligned} h_t &= \text{LSTM}(\text{state}_t, h_{t-1}), \quad t \in [1, n], \\ F &= W_f h_t + b_f, \\ p\left(\frac{y_i}{x_k}\right) &= \text{softmax}(F) = \frac{\exp(F_i)}{\sum_{j=1}^m \exp(F_j)}. \end{aligned} \quad (2)$$

Among them, in addition to receiving the current state information state_t , the LSTM network also receives the LSTM neuron information h_{t-1} at the previous moment. After outputting h_t , it passes through the fully connected layer to obtain a vector F of length two, and finally, the action probability distribution is output through the softmax function.

It is worth noting that the input state of LSTM is the last vector used for classification in the RDM, and there are two output actions:

- (1) Continue: It means that the current information is not enough to determine whether it is a rumor, and let the RDM read another reply message.
- (2) Terminate: It indicates the end of the detection process and outputs the detection result. In other words, the RDM has sufficient information to judge whether the original tweet is a rumor and outputs the result in advance to achieve the purpose of early detection.

The reward function is the core of reinforcement learning. The quality of its design can directly determine the performance of the model. We design the following reward function:

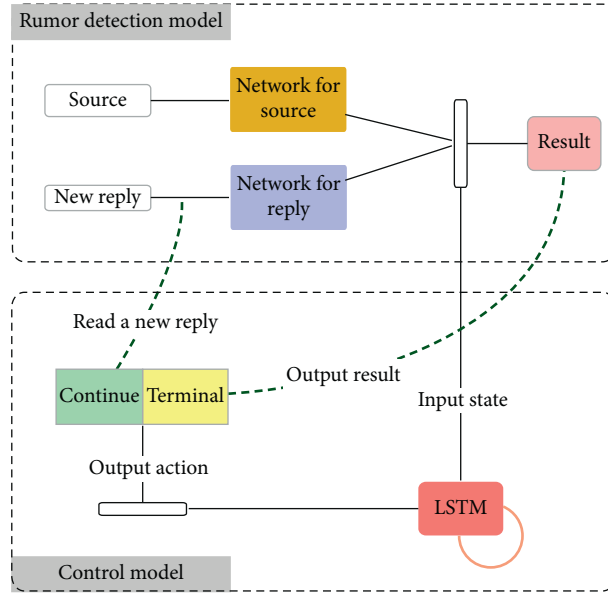


FIGURE 2: Architecture diagram of early rumor detection (ERD) model based on DRQN.

$$r_i = \begin{cases} R, & \text{terminal with correct prediction,} \\ -2P, & \text{terminal with wrong prediction, and label is rumor,} \\ -P, & \text{terminal with wrong prediction, and label is not rumor,} \\ -(\log n + \varepsilon), & \text{continue, } n = 1, 2, 3, \dots \end{cases} \quad (3)$$

Among them, when the model makes a stop-reading action, if the prediction is correct, it will directly get a reward of R to avoid falling into the local optimum; if the prediction is wrong, there are two situations. When the actual label is a rumor, it will receive a $-2P$ punishment; when the actual label is nonrumor, it is punished by $-P$. The reasons for adopting this strategy include two aspects: (1) considering the fact that there are few rumor samples; (2) considering that the losses caused by the rumor detection system are different in the two cases of misjudgment, specifically comparing the effect of misjudging the information that was originally a rumor as not a rumor and the effect of identifying nonrumors as a rumor. Obviously, the former will have a greater impact, because the omission of the rumors by the model will spread the rumors to a greater extent, but for the latter, although the cost of misjudgment has been increased, no rumors have been missed. Therefore, if the information that was originally a rumor is judged to be not a rumor, the model will be punished twice.

When the model continues to read data, it will be punished by $-(\log n + \varepsilon)$, n represents the number of reply messages read by the model, and ε is a small value to avoid the situation where the penalty is 0 when reading the first reply message; the more the response information read, the greater the penalty for continuing to read.

3.2.2. RDM. In view of the difference between the original tweet and the reply information, this paper proposes a dual-engine RDM based on the self-attention mechanism. The

specific model architecture is shown in Figure 3, which is mainly composed of the original tweet network and the reply information network.

In the network for source tweet, the text data passes through the word embedding layer, the GRU network, and the word-level self-attention mechanism in turn. The metadata feature extractor is used to extract the credibility characteristics of the tweet publisher and the basic information of the original tweet. The specific calculation process is as follows:

$$\begin{aligned} x_t &= \text{WordEmbedding}(w_t), \quad t \in [1, n], \\ \text{Output}, h_t &= \text{GRU}(x_t, h_{t-1}), \quad t \in [1, n], \\ \text{Output} &= [\text{output}_1, \text{output}_2, \dots, \text{output}_n], \\ Q &= W_Q \text{Output} + b_Q, \\ K &= W_K \text{Output} + b_K, \\ V &= W_V \text{Output} + b_V, \\ f(Q, K_i) &= Q^T W_a K_i, \\ a_i &= \text{softmax}(f(Q, K_i)) = \frac{\exp(f(Q, K_i))}{\sum_j \exp(f(Q, K_j))}, \\ \text{Attention}(Q, K, V) &= \sum_i a_i V_i. \end{aligned} \quad (4)$$

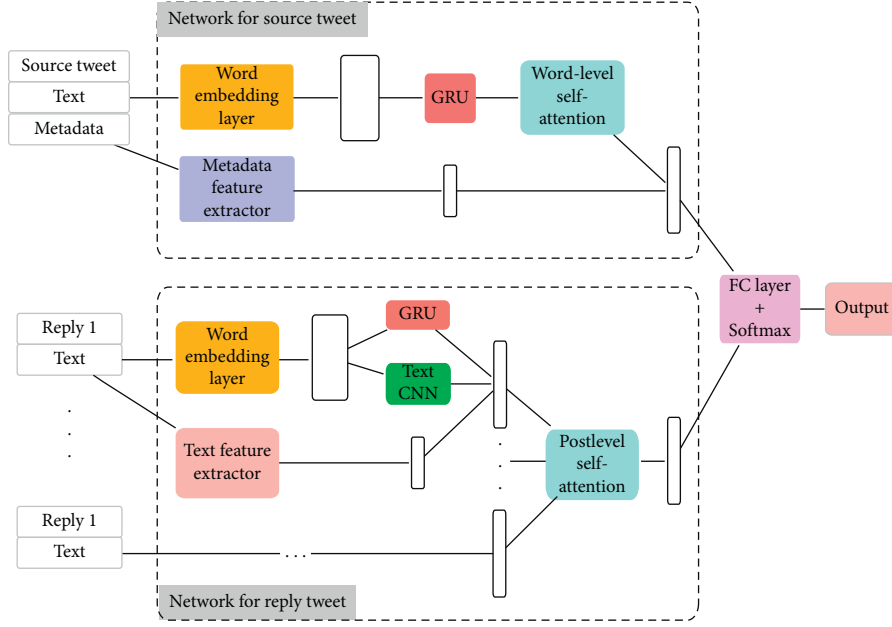


FIGURE 3: Architecture diagram of dual-engine rumor detection model (RDM) based on self-attention mechanism.

The features of metadata are shown in Table 1.

Since the reply information is usually expressed with strong emotional color, the expression is more casual, and there is an unstable word order; this paper considers using two-way GRU, Text-CNN and text feature extractor to extract the reply information features in parallel, and the final feature vector is constructed through vector splicing.

(1) Bidirectional GRU

The word order of reply messages is unstable. In order to extract more information, we use a bidirectional GRU network. The calculation details are shown in the following equation:

$$\begin{aligned} x_t &= \text{WordEmbedding}(w_t), \quad t \in [1, n], \\ \vec{h}_{rt} &= \overrightarrow{\text{GRU}}(x_t, h_{rt-1}), \quad t \in [1, n], \\ \overleftarrow{h}_{lt} &= \overleftarrow{\text{GRU}}(x_t, h_{rt+1}), \quad t \in [n, 1]. \end{aligned} \quad (5)$$

$\overrightarrow{\text{GRU}}$ represents the forward GRU, \vec{h}_{rt} represents the vector representation of forward sequence of words $w_1 \sim w_n$ processed by $\overrightarrow{\text{GRU}}$, and similarly \overleftarrow{h}_{lt} represents the vector representation of reverse sequence of words $w_n \sim w_1$ processed by GRU. At last, \vec{h}_{rt} and \overleftarrow{h}_{lt} are concatenated to obtain the final vector representation of the sentence, as shown in formula

$$h_t = \text{concat}\left(\vec{h}_{rt}, \overleftarrow{h}_{lt}\right). \quad (6)$$

(2) Text-CNN

Aiming at the random features of the way of replying information, this paper uses Text-CNN to extract the semantic features of abnormal word

order. Text-CNN consists of convolutional layer and pooling layer.

The convolutional layer is used to extract text features. The process of extracting text features can be expressed by the following formula:

$$\begin{aligned} a_i &= f(W * M_{i:i+h-1} + b), \\ A &= [a_1, a_2, \dots, a_{n-k+1}]. \end{aligned} \quad (7)$$

$M_{i:i+h-1}$ is the word vector from row i to $i+h$ in the word vector matrix. After performing a linear transformation on $M_{i:i+h-1}$, the activation function f is used to obtain a_i , which represents the i -th text feature extracted by the convolution kernel of length h . Finally, all the features extracted by the convolution kernel are spliced to obtain the vector A . The above process is the processing result of one convolution kernel, and the same steps are repeated for multiple convolution kernels.

In addition, the pooling function of this model adopts the maximum pooling function; that is, after the features extracted by the convolutional layer are obtained, one of the largest features is selected to represent all the features, which can be expressed by the formula

$$\tilde{a} = \max(a_1, a_2, \dots, a_i). \quad (8)$$

(3) Text feature extractor

The text feature extractor can extract relatively intuitive text features, such as statistical negative words, whether there is an exclamation mark, and the similarity with the original tweet, as shown in Table 2.

TABLE 1: Features of metadata.

Feature name	Description
has_url	Whether the tweet contains URL or hyperlink
urls	The number of URLs or hyperlinks contained in the content of the tweet
has_tag	Whether to use the keyword “#” to carry a topic in the content of the tweet
tags	How many topics are carried with the keyword “#” in the content of the tweet
favorite_count	The number of times the tweet was liked
retweet_count	The number of tweets reposted
verified	Is the user authenticated by real name on Twitter?
profile_use_background_image	Whether Twitter users set a background image on their homepage
default_profile_image	Whether the background image set by Twitter users on their homepage is the system default
geo_enabled	Whether users disclose their location information
is_translation_enabled	Whether the user has the translation permission
default_profile	Whether users modify the default personal information
friends_count	Number of users followed
followers_count	User attention
statuses_count	Number of tweets posted by users
description_len	User profile length
favourites_count	Cumulative number of likes of user tweets
listed_count	How many public channels are users involved
user_age	User registration period
coordinates	Does the tweet contain coordinate information
coordinates	Whether the tweet contains URL or hyperlink

TABLE 2: Features of text.

Feature name	Description
word_count	Number of words in the sentence
is_question	Describe whether a question mark appears in the sentence
is_exclamation	Describe whether an exclamation mark appears in the sentence
negation_count	Number of negative words, such as “no, nothing”
badword_count	Number of bad words, such as “fuck, bitch”
similar_to_src	Cosine similarity to the original tweet

The calculation formula of cosine similarity is as follows:

$$\cos(\theta) = \frac{\sum_{i=1}^n (x_i \times y_i)}{\sqrt{\sum_{i=1}^n x_i^2} \times \sqrt{\sum_{i=1}^n y_i^2}} \quad (9)$$

(4) Postlevel self-attention mechanism

The self-attention mechanism at the postlevel refers to the weighting of the self-attention mechanism after feature extraction of all response information so that the model can pay attention to the useful response information. The specific calculation process is shown in the following formula:

$$R = [\text{reply}_1, \text{reply}_2, \dots, \text{reply}_n], \quad (10)$$

$$Q = W_Q R + b_Q, \quad (11)$$

$$K = W_K R + b_K, \quad (12)$$

$$V = W_V R + b_V. \quad (13)$$

In formula (10), reply_n represents the encoding of a reply message by the reply message network. R is

the set of reply message codes. The self-attention mechanism requires three vectors Q , K , and V to represent query, key, and value, respectively, which are obtained from the vector R through three different linear transformations. Attention can be calculated based on these vectors. The specific calculation process is shown in the following formula:

$$f(Q, K_i) = Q^T W_a K_i, \quad (14)$$

$$a_i = \text{softmax}(f(Q, K_i)) = \frac{\exp(f(Q, K_i))}{\sum_j \exp(f(Q, K_j))}, \quad (15)$$

$$\text{Attention}(Q, K, V) = \sum_i a_i V_i. \quad (16)$$

In formula (14), $f(Q, K_i)$ is a general linear transformation to calculate the similarity between Q and K . The weight a_i that needs attention for each post is obtained through the softmax function, and the product of the weight and the vector V is expressed as the response information processed by attention.

3.3. *Model Building.* In the process of building an ERD model, pretraining RDM and time difference method are used to train CTM (control model).

(1) Pretraining RDM

There must be a reliable RDM as a basis before training CTM. This pretrained RDM reads all the response information during the training process. This also means that the performance of the ERD finally obtained after joining the CTM will not be higher than the performance of the pretrained RDM. The significance of adding CTM is how to make RDM get the best performance with the least information, and its best performance is that of the pretrained RDM.

In the RDM training process, the batch size is 80, the dropout is 0.4, and the loss function is cross-entropy loss function. In addition, Adam optimizer is used in the model and the initial learning rate is 0.0005. In the training process, when the model's loss on the validation set does not decrease twice in a row, the learning rate is reduced by 10 times. When the model's loss on the validation set stabilizes, the training process stops.

(2) Training CTM

If CM is trained directly, its parameters will lack stability, which will make it difficult for the model to converge. Therefore, in order to speed up the convergence, we use a dual network structure and experience playback mechanism to train CTM.

Dual networks are two CTM networks with the same structure, which are called: current network (parameter θ) and target network (parameter θ'), respectively. The experience pool stores n four-tuple records (s_t, a_t, r_t, s_{t+1}) about the environment, and a batch of samples are randomly taken from the experience pool for training each time. The training process is to train the current network first and update the target network with the parameters of the current network when a certain batch is reached.

Because the rumor detection problem is relatively special, traditional reinforcement learning is aimed at an environment, such as a game scene, but in the rumor detection problem, each rumor data is actually an environment, so it is necessary to consider multiple environments in the process of constructing the experience database. For environmental factors, the specific training process is shown in Algorithm 1.

4. Experiments Results and Analysis

4.1. *Experimental Environment.* The experimental environment is shown in Table 3.

4.2. *Dataset.* The experimental datasets include the public rumor dataset PHEME Dataset of Rumors and Non-Rumors (PHEME Rumor) [7], which is rumors and nonrumors data

about five breaking news events collected on Twitter by ArkaitzZubiaga in 2016. The data distribution of each breaking news event is shown in Table 4.

The dataset is divided into training dataset, verification dataset, and test dataset with the ratio of 7:1:2. The validation set is used to observe the real-time training results. Based on the best training results, the best model is selected as the final model. Finally, the test set is used to test the performance of the model.

4.3. *Baselines.* This paper selects the following RDM as the baseline algorithm:

- (1) CRF: the RDM proposed by Zubiaga et al. [23].
- (2) GAN-GRU: the RDM proposed by Ma et al. [24].
- (3) RDM: the RDM in the model proposed by Zhou et al. [23].
- (4) LSTM: an LSTM network for rumor veracity proposed by Singh et al. [25].
- (5) LSTM-Attention: attention-based LSTM network for rumor veracity proposed by Singh et al. [26].
- (6) SA-SE: the model proposed in this paper uses only a single-engine model using a word-level attention mechanism (only the original information network).
- (7) SA-DE: the model proposed in this paper only uses a dual-engine model using sentence-level attention mechanism.
- (8) DSA-DE: the model proposed in this paper is based on the DSA dual-engine model.

In the current ERD research, only the ERD [21] uses reinforcement learning to solve the ERD problem, so we consider using ERD as a baseline method. The control variable method is used in the comparison link, and the submodels of the two models are cross-combined into multiple models for comparison experiments.

First, the reinforcement learning module in ERD is named RL1, and the RDM is named RDM1; the reinforcement learning module in the model proposed in this paper is named RL2, and the RDM is named RDM2. Finally, the experimental models to be compared are divided into the following three models:

- (1) RDM1_RL1: ERD model
- (2) RDM2_RL1: the model proposed in this paper only contains the RDM and the control module in the ERD
- (3) RDM2_RL2: the ERD model proposed in this paper

5. Experimental Results and Analysis

5.1. *Rumor Detection Performance Evaluation.* Comparative experimental results of eight RDM are shown in Table 5.

It can be seen from Table 5 that the GAN-GRU performs well in terms of accuracy, but it has poor performance in terms of precision and recall; the model of Zhou et al. is

```

Input: Network  $Q(s, a)$ , Environment set  $E$ , Experience pool  $P$ 
Output:  $Q(s, a, \theta')$ 
(1) Initialize current network  $Q(s, a, \theta)$ , and target network  $Q(s, a, \theta')$ ,  $\theta' = \theta$ 
(2) for each epoch do
(3)   Select an environment  $e$  from  $E$ 
(4)   Initialize environment  $e$ , and get state  $s_t$ 
(5)   while true do
(6)     According to  $s_t$ , use  $\epsilon$ -greedy strategy to select action  $a_t$  from  $Q(s, a, \theta)$ 
(7)     Perform action  $a_t$  in the environment to get the new state  $s_{t+1}$  and reward  $r_t$ 
(8)     if  $P$  is full do
(9)       Delete the oldest experience record
(10)    end if
(11)    Insert  $(s_t, a_t, r_t, s_{t+1})$  into  $P$ 
(12)     $s_t \leftarrow s_{t+1}$ 
(13)    if  $s_t$  is the last state do
(14)      break
(15)    end if
(16)  end while
(17)  if  $P$  is full do
(18)    Select a batch of records from  $P$  randomly
(19)  for each record do
(20)    Use target network to get  $y_t = r_t + \gamma \max_{a_{t+1}} Q(s_{t+1}, a_{t+1}, \theta')$ 
(21)    Use loss function  $(y_t - Q(s_t, a_t, \theta))^2$  to update current network  $Q(s, a, \theta)$ 
(22)    Update current network with target network every  $n$  epochs
(23)  end for
(24) end if
(25) end for

```

ALGORITHM 1: Training DRQN.

TABLE 3: Experimental environment.

Name	Version
CPU	Intel(R) Xeon(R) Platinum 8160 CPU @ 2.10 GHz
GPU	NVIDIA Tesla P4 8 GB GPU
OS	Centos7.4
Memory	4 GB
PyTorch	1.7.1

relatively stable; the LSTM-Attention achieved the highest score in precision; and the models proposed in this paper are SA-SE, SA-DE, and DSA-DE that have achieved good results in all four indicators. When the model adds the double self-attention (DSA) mechanism and the dual-engine network, the DSA-DE is compared to the sentence-level dual-engine network model SA-DE, and the single-level attention and single-engine network model SA is compared with SE; the accuracy rate is increased by 2.3% and 5%, respectively. It shows that the dual-engine network and DSA mechanism have improved the performance of the rumor detection task.

Figure 4(a) describes the change trend of the model SA-SE, SA-DE, DSA-DE loss value for the validation set during the training process. It can be seen from the figure that the DSA-DE model achieved the lowest loss. Although the initial loss of the SA-DE model that does not use the sentence-level attention mechanism decreases the fastest, the final training effect is not as good as the DSA-DE model that uses the DSA mechanism. It can be seen that both the DSA mechanism and the dual-engine network can improve the learning ability of the model.

Figure 4(b) shows the change trend of the model SA-SE, SA-DE, DSA-DE accuracy for the validation set during the training process. It can be seen that the DSA-DE model has the fastest learning ability. After 10 rounds of training, the accuracy of the model has stabilized at around 85%. Although the SA-DE model that does not use word-level attention has strong initial learning ability, the final learning result does not exceed the DSA-DE model. Therefore, both the DSA mechanism and the dual-engine network can improve the learning ability of the model.

5.2. ERD Efficiency Evaluation. Since we utilize reinforcement learning theory to achieve the purpose of ERD by controlling the number of replies input. In order to evaluate the early nature of the model, firstly, we use the standard indicators such as accuracy, precision, recall, and F_1 score.

It can be seen from Table 6 that when the control module RL1 is added, the accuracy of RDM2_RL1 is 0.05 higher than that of RDM1_RL1, indicating that the performance of the dual-engine RDM based on the DSA mechanism proposed in this paper is better than the rumor in ERD Detection module. Comparing the control modules, when RDM2 is added to the control modules RL1 and RL2, the accuracy rates drop to 0.80 and 0.81, respectively, indicating that the DRQN-based control module proposed in this paper can maintain a high accuracy rate.

Secondly, we use the average number of responses for each sample as one of the evaluation indicators, which is recorded as mean posts used. In order to more intuitively

TABLE 4: Rumor dataset distribution.

Event	Number of rumors (proportion)	Number of rumors (proportion)
Charlie Hebdo	456 (22.0%)	1621 (78.0%)
Ferguson	284 (24.8%)	859 (75.2%)
Germanwings Crash	238 (50.7%)	231 (49.3%)
Ottawa Shooting	470 (52.8%)	420 (47.2%)
Sydney Siege	522 (42.8%)	699 (57.2%)
Sum	1970 (34.0%)	3830 (66.0%)

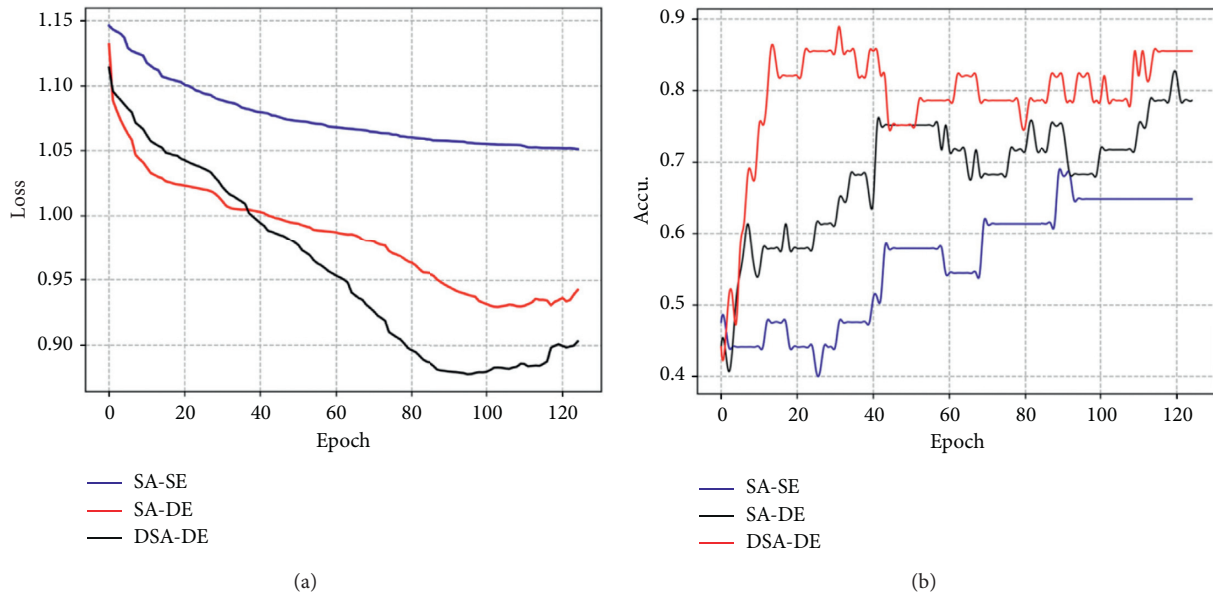


FIGURE 4: Comparison of RDM during training. (a) Loss value trend analysis. (b) Accuracy analysis.

TABLE 5: Experimental results of RDM.

Model name	Accuracy	Precision	Recall	F_1 score
CRF	0.67	0.67	0.56	0.60
GAN-GRU	0.78	0.53	0.35	0.42
RDM	0.77	0.74	0.74	0.74
LSTM	0.81	0.77	0.69	0.72
LSTM-Attention	0.83	0.83	0.79	0.81
SA-SE	0.78	0.70	0.73	0.71
SA-DE	0.81	0.79	0.80	0.80
DSA-DE	0.84	0.81	0.82	0.82

reflect the performance of the model in terms of accuracy and timeliness, we propose an evaluation indicator called early detection rate:

$$\text{early detection rate} = \frac{F_1\text{-Score} * 10}{\text{mean posts used}}. \quad (17)$$

Figure 5 shows the experimental results of the models RDM1_RL1, RDM2_RL1, and RDM2_RL2 with the control module added in the early detection. The average number of reply messages used and the early detection rate are shown in Figures 5(a) and 5(b), respectively. It can be seen that the model RDM2_RL2 proposed uses the least amount of information. On average, each piece of data uses only 1.004

reply messages and has an early detection rate of 8.058, indicating that the model proposed in this paper can find a better balance between accuracy and timeliness so that the model can identify the rumors in the early stage while ensuring the accuracy.

Figure 6 shows the change of the average reward value of the model RDM2_RL2 during the training process. It can be seen that the average reward value of the model is stable between 23 and 24 after 40 rounds of training, indicating that the DRQN-based control module proposed is effective.

Figure 7 shows the changes in the early detection rate of models RDM1_RL1, RDM2_RL1, and RDM2_RL2. It can be seen that, compared to the model RDM1_RL1, the model RDM2_RL1 has the same learning ability, but the final result

TABLE 6: Experimental results of ERD model.

Model name	Accuracy	Precision	Recall	F_1 score
RDM1_RL1 (ERD)	0.75	0.73	0.73	0.75
RDM2_RL1	0.80	0.78	0.78	0.80
RDM2_RL2	0.81	0.79	0.79	0.81

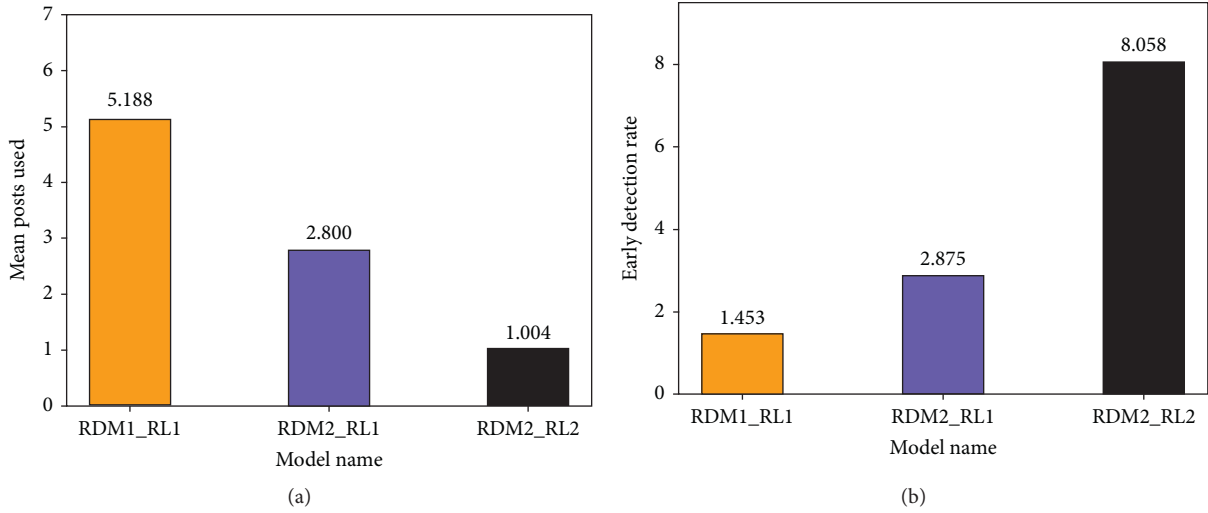


FIGURE 5: ERD comparison experiment results. (a) Mean posts used. (b) Early detection rate.

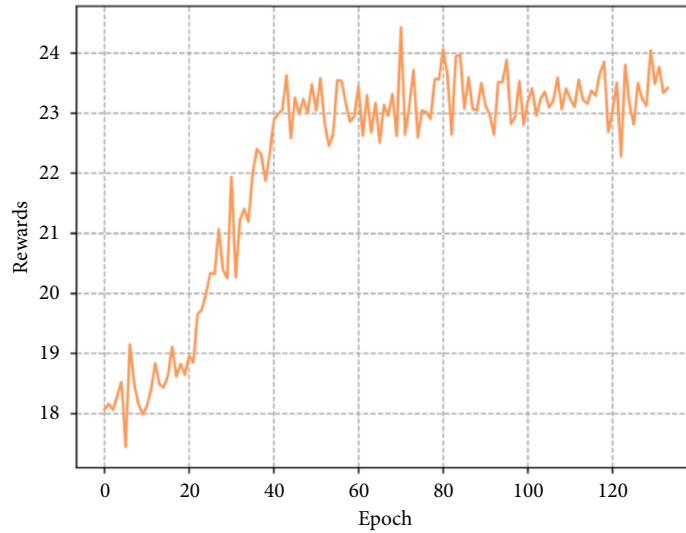


FIGURE 6: Reward value changing state during training.

of RDM2_RL1 training is better than RDM1_RL1. It shows that the RDM proposed is more helpful to detect rumors. The model RDM2_RL2 proposed has a stronger learning ability. It reaches the peak of early detection rate at about 50

rounds of training, and the early detection rate far exceeds RDM1_RL1 and RDM2_RL1. Therefore, the model proposed has good performance in the accuracy and timeliness of rumor detection.

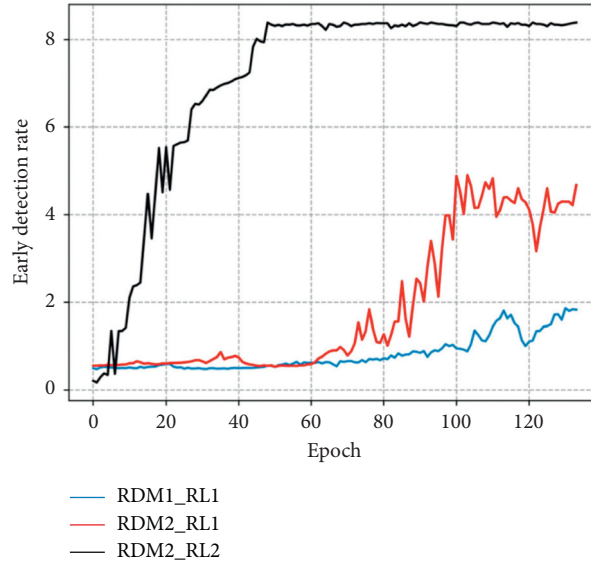


FIGURE 7: Early detection rate changing state during training.

6. Conclusions

In terms of rumor detection, this paper first analyzes the existing research on three problems: the inability to obtain the optimal representation of the reply information, the ignorance of the difference between the original tweet and the reply information in the tweet, and the inability to handle redundant data well. In response to the above problems, this paper uses the difference between the original tweet and the reply information in the Twitter data as an entry point and proposes a dual-engine RDM, which separately deals with the original tweet and the reply information; on the remaining problem, the DSA mechanism is proposed to solve the problem of data redundancy in the two dimensions of sentences and words. For the existing multitask model, there is a problem that the optimal representation of the reply information cannot be obtained. This paper uses a single-task learning model. Let the model itself learn to encode the reply message. The final experimental results show that the method proposed in this paper has a better detection effect.

In terms of ERD, this paper considers solving the problem of ERD from the perspective of reinforcement learning. First, the following problems are found through analysis of existing research: the potential meaning of the state sequence is ignored, the reward function is imperfect, and the performance of the RDM is poor. In response to the above problems, this paper proposes an ERD model based on DRQN and describes the model in detail. In order to analyze the experimental results more effectively, this paper proposes the evaluation index of ERD rate to evaluate the performance of the ERD model. Finally, this paper is verified on the rumor dataset. The experimental results show that this paper can detect rumors earlier under the premise of ensuring the accuracy of rumor detection.

Although the model proposed in this paper has achieved good results by comparing the baseline method, it can still be optimized from the following perspectives.

In natural language processing tasks, the data cleaning stage cannot be ignored. In future research, more fine-grained methods can be considered to clean information such as words, sentences, special symbols, and URLs.

In terms of rumor detection, the language model can be modeled using the relatively new BERT. BERT can learn specific expressions of words in specific language scenarios, and the model may achieve better performance. In addition, in terms of features, more meaningful features can be explored for experimentation.

In the ERD problem, you can use more powerful reinforcement learning algorithms such as A3C to model. The reward function and training method can also be further optimized. In addition, some potential features in time can be explored for modeling. Of course, the problem of ERD is not necessarily limited to reinforcement learning, and there may be more suitable methods for ERD.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was supported by the NSFC-Xinjiang Joint Fund Key Program (Grant no. U2003206) and the National Natural Science Foundation of China (Grant no. 61972255).

References

- [1] G. Q. Sun, W. P. Shi, and L. Wang, "A review of group behavior in online social networks and future prospects," *Journal of Modern Information*, vol. 36, no. 2, pp. 38–42, 2016.
- [2] We Are Social, Global Digital Report, 2019, <https://wearesocial.com/global-digital-report-2019>.
- [3] T. Wang, Y. Lu, J. Wang, H.-N. Dai, X. Zheng, and W. Jia, "EIHDP: edge-intelligent hierarchical dynamic pricing based on cloud-edge-client collaboration for IOT systems," *IEEE Transactions on Computers*, p. 1, 2021.
- [4] T. Wang, Y. X. Mei, X. X. Liu et al., "Edge-based auditing method for data security in resource-constrained Internet of Things," *Journal of Systems Architecture*, vol. 114, Article ID 101971, 2021.
- [5] R. H. Knapp, "A psychology of rumor," *Public Opinion Quarterly*, vol. 8, no. 1, pp. 22–37, 1944.
- [6] M. Al-Sarem, W. Boulila, M. Al-Harby et al., "Deep learning based rumor detection on microblogging platforms: a systematic review," *IEEE Access*, vol. 7, pp. 152788–152812, 2019.
- [7] A. Zubiaga, M. Liakata, and R. Proctor, *PHEME Dataset of Rumours and Non-Rumours Dataset*, https://figshare.com/articles/PHEME_dataset_of_rumours_and_non-rumours/4010619, 2016.
- [8] C. Castillo, M. Mendoza, and B. Poblete, "Information credibility on Twitter," in *Proceedings of the 20th International Conference on World Wide Web*, pp. 675–684, Hyderabad, India, 2011.
- [9] S. Kwon, M. Cha, K. Jung et al., "Prominent features of rumor propagation in online social media," in *Proceedings of the IEEE 13th International Conference on Data Mining*, pp. 1103–1108, Dallas, TX, USA, 2013.
- [10] S. Vosoughi, *Automatic Detection and Verification of Rumors on Twitter*, Massachusetts Institute of Technology, Cambridge, MA, USA, 2015.
- [11] K. Wu, S. Yang, and K. Q. Zhu, "False rumors detection on sina-weibo by propagation structure," in *Proceedings of the 2015 IEEE 31st International Conference on Data Engineering*, pp. 651–662, Seoul, South Korea, April 2015.
- [12] Y. Wu, H. Huang, Q. Wu, A. Liu, and T. Wang, "A risk defense method based on microscopic state prediction with partial information observations in social networks," *Journal of Parallel and Distributed Computing*, vol. 131, pp. 189–199, 2019.
- [13] J. Ma, W. Gao, Z. Wei et al., "Detect rumors using time series of social context information on microblogging web-sites," in *Proceedings of the 24th ACM International Conference on Information and Knowledge Management*, pp. 1751–1754, Melbourne, Australia, October 2015.
- [14] J. Yu, J. Jiang, L. Min et al., "Coupled hierarchical transformer for stance-aware rumor verification in social media conversations," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, pp. 1392–1401, November 2020.
- [15] J. Ma, W. Gao, and K. F. Wong, "Detect rumor and stance jointly by neural multi-task learning," in *Proceedings of the 2018 Companion: The 2018 Web Conference Companion*, pp. 585–593, Lyon, France, April 2018.
- [16] Q. Li, Q. Zhang, and L. Si, "Rumor detection by exploiting user credibility information, attention and multi-task learning," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 1173–1179, Florence, Italy, July 2019.
- [17] J. Ma, W. Gao, and K. F. Wong, *Rumor Detection on Twitter with Tree-Structured Recursive Neural Networks*, Association for Computational Linguistics, Stroudsburg, PA, USA, 2018.
- [18] S. Kumar and K. M. Carley, "Tree LSTMS with convolution units to predict stance and rumor veracity in social media conversation," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 5047–5058, Florence, Italy, July 2019.
- [19] A. Gupta, P. Kumaraguru, C. Castillo, and P. Meier, "TweetCred: real-time credibility assessment of content on twitter," in *Proceedings of the International Conference on Social Informatics*, pp. 228–243, Barcelona, Spain, November 2014.
- [20] S. Dungs, A. Aker, N. Fuhr et al., "Can rumour stance alone predict veracity?" in *Proceedings of the 27th International Conference on Computational Linguistics*, pp. 3360–3370, Santa Fe, NM, USA, August 2018.
- [21] K. Zhou, C. Shu, B. Li et al., "Early rumour detection," in *Proceedings of the 2019 Conference of the North*, Minneapolis, MN, USA, 2019.
- [22] X. Wang, T. Zhang, and Y. G. Jin, "Overview of anomaly detection algorithms," *Modern Computer*, no. 30, pp. 21–26, 2020.
- [23] A. Zubiaga, M. Liakata, and R. Proctor, "Exploiting context for rumour detection in social media," in *Proceedings of the International Conference on Social Informatics*, pp. 109–123, Oxford, UK, September 2017.
- [24] J. Ma, W. Gao, and K. F. Wong, "Detect rumors on Twitter by promoting information campaigns with generative adversarial learning," in *Proceedings of the World Wide Web Conference*, pp. 3049–3055, Geneva, Switzerland, October 2019.
- [25] J. P. Singh, P. R. Nripendra, and Y. K. Dwivedi, "Rumour veracity estimation with deep learning for Twitter," in *Proceedings of the International Working Conference on Transfer and Diffusion of IT*, Accra, Ghana, 2019.
- [26] J. P. Singh, A. Kumar, N. P. Rana et al., "Attention-based LSTM network for rumor veracity estimation of tweets," *Information Systems Frontiers*, pp. 1–16, 2020.

Research Article

STQ-SCS: An Efficient and Secure Scheme for Fine-Grained Spatial-Temporal Top- k Query in Fog-Based Mobile Sensor-Cloud Systems

Jie Min ¹, Junbin Liang ², Xingpo Ma ³, and Hongling Chen ⁴

¹School of Information Engineering, Xinyang Agriculture and Forestry University, Xinyang 464000, Henan, China

²School of Computer and Electronics Information, Guangxi University, Nanning 530004, Guangxi, China

³School of Computer and Information Technology, Xinyang Normal University, Xinyang 464000, Henan, China

⁴Guangdong Polytechnic of Science and Technology, Zhuhai 519090, Guangdong, China

Correspondence should be addressed to Xingpo Ma; maxingpo@xynu.edu.cn

Received 11 March 2021; Revised 5 May 2021; Accepted 19 May 2021; Published 29 May 2021

Academic Editor: Lu Liu

Copyright © 2021 Jie Min et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

With the emergence of the fog computing and the sensor-cloud computing paradigms, end users can retrieve the desired sensory data generated by any wireless sensor network (WSN) in a fog-based sensor-cloud system transparently. However, the fog nodes and the cloud servers may suffer from many kinds of attacks on the Internet and become semitrusted, which threatens the security of query processing in the system. In this paper, we investigated the problem of secure, fine-grained spatial-temporal Top- k query in fog-based mobile sensor-cloud systems (FMSCSs) and proposed a novel scheme named STQ-SCS to tackle the problem based on the virtual grid construction and the size-order encryption-binding techniques. STQ-SCS can preserve the privacy of the sensed data items and their scores and make end users verify the completeness of the query results of fine-grained spatial-temporal Top- k queries with a 100% successful rate even if the fog nodes and the cloud servers are not totally trustworthy. Besides the good security performance, simulation results indicate that STQ-SCS is also an efficient scheme that incurs a much lower communication cost than the state-of-the-art schemes on securing fine-grained spatial-temporal Top- k query in FMSCSs.

1. Introduction

As one important component of Internet of Things (IoT) [1], wireless sensor networks (WSNs) [2] can be used in many application scenarios and are still being studied [3] by many researchers even though extensive research has been carried out on WSNs for the past two decades. However, traditional WSNs are usually *single-user centric* [4], where a user deploys and owns its own WSN and another party is not able to access the sensed data generated by such a WSN. To remedy this shortcoming, researchers have conceived a new paradigm, namely, the *sensor-cloud* paradigm [5–7], in recent years. A typical sensor-cloud model is shown in Figure 1(a), where the sensor-cloud architecture serves as the intermediate stratum between the end users and the physical sensor nodes [4]. However, early sensor-cloud architectures are still

not perfect, and they encounter many new challenges, such as providing real-times services and efficiently managing the physical sensor nodes. In [8], a new sensor-cloud architecture, namely, the fog-based sensor-cloud framework, was proposed, and the basic model of the fog-based sensor-cloud framework is shown in Figure 1(b). The main difference between early sensor-cloud architectures and the fog-based sensor-cloud framework is that the latter has a fog layer while the former does not have. The fog layer is mainly composed of fog nodes, which can fuse and store the collected sensed data, respond to real-time applications, and efficiently manage the physical sensor nodes [8]. In the fog-based sensor-cloud framework, end users can not only retrieve the sensed data items, which they are interested in directly from the nearby fog nodes, but also obtain the shared sensed data from the cloud by sending queries to the

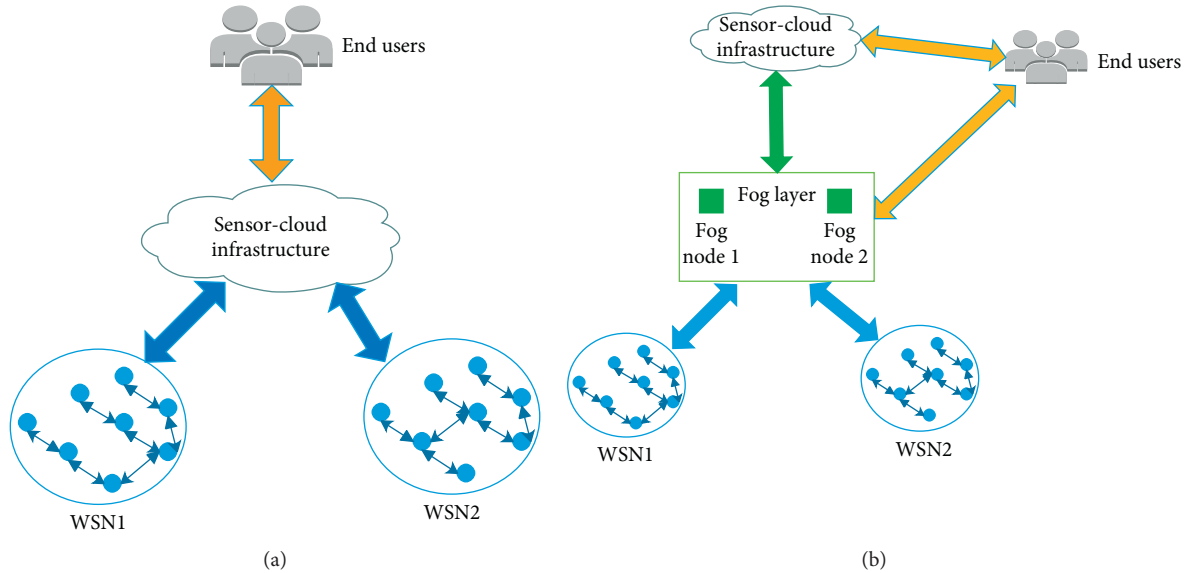


FIGURE 1: The sensor-cloud architecture (generally, there are more than 2 WSNs in real applications, and we just use 2 WSNs as representatives). (a) Common sensor-cloud architecture. (b) Fog-based sensor-cloud architecture.

cloud if there are no data which they want in the near fog nodes.

Although the fog-based sensor-cloud framework brings a lot of benefits as described in [8], it encounters many potential security threats. The fog nodes may be captured by the nearby attackers or may suffer from the attacks arising from the cloud. In other words, the fog nodes may become untrusted [9, 10] under such attacks. Meanwhile, the application servers in the cloud are facing many kinds of attacks, and some of the cloud servers may also not be trustworthy [11–13]. Under this background, how to ensure the integrity and the confidentiality of the sensed data items retrieved by the end users in the fog-based sensor-cloud systems is a thorny-and-burning problem. Such a problem is much more challenging in fog-based mobile sensor-cloud systems (FMSCSs), where the sensor nodes are mobile, considering that the sensed data retrieved by end users must satisfy the spatial-temporal requirements of the queries launched by end users.

In this paper, we focus on fine-grained spatial-temporal Top- k queries and make efforts to tackle the above-mentioned problem. The concept of fine-grained spatial-temporal Top- k queries is defined in Definition 1 in Section 3. In a word, a fine-grained spatial-temporal Top- k query refers to a query that tries to find out the top k sensed data items generated in a specific time interval and a specific region of a specific WSN deployment field. To our best knowledge, there is no work studying the problem of secure fine-grained spatial-temporal Top- k query in fog-based sensor-cloud systems at present. In brief, the main contributions of this paper are twofold:

- (i) It studies the problem of secure fine-grained spatial-temporal Top- k query in FMSCSs and proposes a novel scheme named STQ-SCS to ensure the integrity and confidentiality of the sensed data items

retrieved by end users. It provides sound theoretical analysis on the security of STQ-SCS. According to the analysis, STQ-SCS is not only able to preserve the privacy of the sensed data items retrieved by end users but also detect the incomplete query results successfully for fine-grained spatial-temporal Top- k query under the security model presented in this paper.

- (ii) Extensive simulations were conducted in the paper, and the results show that STQ-SCS is much more efficient than the related state-of-the-art schemes.

The remainder of this paper is organized as follows. Section 2 summarizes the related schemes; Section 3 describes the system model, the security model, the definitions of some terminologies, and the problem statement; Section 4 presents the proposed scheme STQ-SCS in detail; Section 5 analyzes the security of STQ-SCS; In Section 6, STQ-SCS is compared with the related state-of-the-art schemes through extensive simulations; Section 7 provides performance evaluation. Section 8 concludes this study.

2. Related Works

Since there is no work about secure fine-grained spatial-temporal Top- k query in FMSCSs at present, we mainly investigate the related works in Cloud Computing, Two-tiered Wireless Sensor Networks (TWSNs), and Two-tiered Mobile Wireless Sensor Networks (TMWSNs) in this section.

2.1. Securing Top- k Queries in Cloud Computing. Top- k queries in the cloud are generally securely processed based on the data that are outsourced on cloud servers by the same data owner. In Cloud Computing, the data owner knows all its outsourced data and thus can construct the encrypted

data structure, such as EHL [14], the binary heap [15], or other tree-like structures [16–18], based on the whole data set to facilitate Top- k query without losing data privacy, while in FMSCSs, expect for the fog nodes that are considered as not fully trusted, each sensor node just knows only a small part of the whole data generated by the WSN where it is located, and it thus cannot construct the encrypted data structure of the whole data before outsourcing its data to a fog node or the cloud.

Moreover, existing schemes proposed for secure Top- k query in Cloud Computing are based on the strong processing ability and rich resources of the cloud servers, and they never consider the resource-limited sensor nodes which are also weak in computing. Thus, they are not fit for FMSCSs.

2.2. Securing Top- k Queries in TWSNs. The study of securing Top- k queries in TWSNs was originally launched by the authors in [19], where three schemes are proposed to preserve the completeness of the Top- k query results in TWSNs. The three schemes were proposed based on the MAC (Message Authentication Code) technique, which requires each sensed data item to be attached with an MAC as its proof data. Then, many other schemes that use a similar technique appeared, such as those in [19–24]. However, the MAC-based technique is relatively less efficient because attaching an MAC to each sensed data item brings large quantity of extra data since a MAC takes almost 40% of the volume of a sensed data item according to [19].

Besides the MAC-based technique, some other methods were also proposed to ensure the privacy of the sensed data and the completeness of the Top- k query results in TWSNs, such as inserting digital watermarks or dummy readings into the normal ones [25] and constructing data aggregation trees [26, 27]. However, inserting digital watermarks or dummy readings into the measure data makes it hard and complicated for the users to extract the normal readings from the hybrid ones, and it also brings a lot of redundant data, which further leads to the increase of the communication cost of both the sensor nodes and fog nodes.

What is more, one of the most important common points of these schemes is that they are all proposed for TWSNs where nodes are static [28], and they cannot perfectly treat the security threats faced by spatial-temporal Top- k query in FMSCSs, where attackers can launch much more covert attacks. When a mobile sensor node travels from the queried region to other regions or vice versa in the queried time interval, some sensed data generated by the sensor node may be in the queried region, and others may not. Obviously, the sensed data generated out of the queried region by the traveling sensor node are not the qualified ones that satisfy the requirements of the spatial-temporal Top- k query. However, few securing Top- k query schemes proposed in TWSNs consider this, which leaves leaks for the attackers to launch new kinds of covert attacks. For example, the attackers may replace the data items that are generated in the queried region by a sensor node with those produced out of the queried region by the same sensor node.

2.3. Securing Top- k Queries in TMWSNs. The first work on securing Top- k queries in TMWSNs was done by Liu et al. in 2015 [29], when they presented a novel network architecture, namely, TMWSNs, and proposed a scheme VTMSN to ensure the completeness of spatial-temporal Top- k query in TMWSNs. The main techniques used in VTMSN are symmetric encryption and information binding. Specifically, it binds the score of each sensed data item with its corresponding generation time, location, and value ranking order by concatenating and encrypting them with the kept symmetric key. Although VTMSN increases the difficulty for the attackers to undermine the completeness of the query results because of the binding relationships, it still has shortcomings. One is that it cannot preserve the privacy of the sensed data items since it leaves the data items disclosed to the fog nodes for ease of Top- k query processing on them; another one is that there should be a large volume of location data transported together with the sensed readings, which greatly increases the communication cost of the sensor nodes and fog nodes.

To overcome the latter shortcoming of VTMSN, Wu et al. proposed a scheme named EVTopk [30] in 2016. EVTopk achieves completeness preservation of the Top- k query results by using the HMAC (Hash Message Authentication Code), which is formed by making hashing and encryption operations on the concatenated items including the score, the location, and the neighboring HMAC. However, since each sensed data item should be attached with an HMAC in EVTopk, the HMACs account for a large proportion of the data reports of the sensor nodes and the query results. Moreover, EVTopk is not able to achieve data privacy preservation either. In [31], a comparative study was made on the two schemes, EVTopk and VTMSN. To further decrease the volume of the proof data in the data reports and the query results, in 2018, a scheme named VIP-TQ was proposed to preserve the integrity of the query results for spatial-temporal Top- k query in TMWSNs. In VIP-TQ, sensed data are bound together with their location as well as their neighboring data score using pairwise-key-based encryption. Although the binding can effectively prevent the compromised fog nodes from undermining the integrity of the Top- k query results, it leaves the scores of the sensed data disclosed to the storage nodes, which increases the risk of divulging the privacy of the sensed data. In the same year, Ma et al. proposed two other schemes, namely, SSSTQ1 and SSSTQ2 [32], for securing spatial-temporal Top- k in TMWSNs. However, a large number of original locations associated with the sensed data items are added into the data reports and the query results for integrity verification, which heavily increases the communication cost of the systems.

In summary, although there are many schemes related to secure Top- k query in existing works, they either have obvious shortcomings or cannot be used in FMSCS, which motivates us to do further work in this paper.

3. Models, Notations, and Problem Statement

3.1. System Model. The system model of FMSCSs is shown in Figure 2. In the model, TA is short for trusted authority [33],

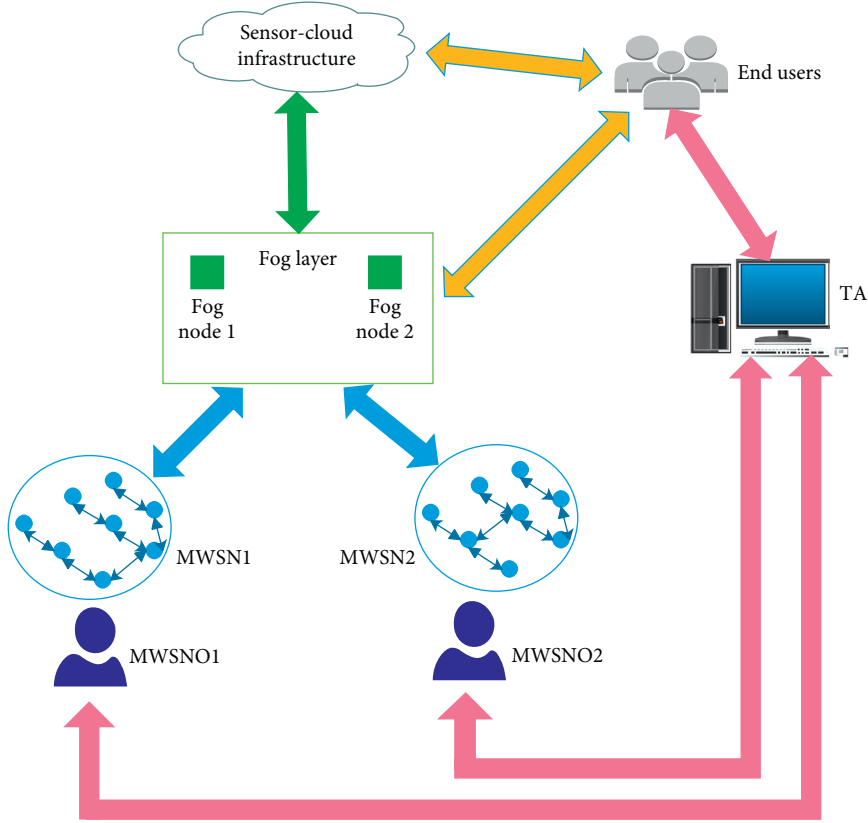


FIGURE 2: System model of FMSCSs.

which is a trustworthy party. TA is used to authenticate the identity of end users and MWSNOs (Mobile Wireless Sensor Network Owners) and distribute the secret keys to them. Each fog node in the fog layer connects and manages one MWSN (Mobile Wireless Sensor Network), and each MWSN is assumed to be composed of N mobile sensor nodes and is owned by a MWSNO. Specifically, the main responsibility of each fog node is as follows: (1) Collecting, processing, and storing the sensed data items updated by the sensor nodes in its corresponding WSN; (2) managing the mobile sensor nodes in its corresponding MWSN; and (3) responding to the queries that may be sent from the Cloud or the end users directly. End users can retrieve the desired data by launching and sending queries to the cloud or the fog nodes directly if they are not far from the fog nodes. If a cloud server receives a query from some end user, it first determines the fog node, which satisfies the region requirement of the query, and then sends the query to the fog node; if a fog node receives a query, it processes the query locally and sends the query result to the party (the cloud or the end user) who has sent the query.

The mobile sensor nodes in WSNs periodically upload their sensed data to the corresponding fog nodes in the fog layer. We divide time into epochs, and take the time length of each epoch as the period for each sensor node to upload its sensed data items. We assume that mobile sensor nodes in

each WSN do not move all the time. They stay at some target locations for certain time intervals when they reach the positions, and go on moving to other target locations if it is necessary. Moreover, we assume that the mobile sensor nodes only generate sensed data items when they are staying at their target locations. Besides, it is assumed that each mobile sensor node just moves within the WSN field where it is located, since it will cost a lot of energy for the sensor nodes to move among different WSN-deployed fields.

In this paper, we use the set $\{D_{i,j,1}^t, D_{i,j,2}^t, \dots, D_{i,j,\mu_{i,j}^t}^t, D_{i,j,\mu_{i,j}^t}^t, j^t\}$ to denote the sensed data items generated by sensor node S_i at its j^{th} target location in the t^{th} epoch T^t , where $\mu_{i,j}^t$ is the total number of the sensed data items generated by S_i at its j^{th} target location in T^t . For any sensed data item $D_{i,j,x}^t$, its corresponding data score $d_{i,j,x}^t$ can be worked out using a public scoring function $f(\ast)$ [19], namely, $d_{i,j,x}^t = f(D_{i,j,x}^t)$. Without loss of generality, we assume different sensed data items have distinct scores [19]. Moreover, in order to facilitate presentation, we assume that the ranking orders of the sensed data items generated by any sensor node at a target location are consistent with their subscript digital numbers. For example, there is $D_{i,j,1}^t < D_{i,j,2}^t < \dots < D_{i,j,\mu_{i,j}^t-1}^t < D_{i,j,\mu_{i,j}^t}^t$, where i and j are the node ID and the target location ID of S_i , respectively. The specific meanings of the notations used in this paper are listed in Table 1.

TABLE 1: Notations and their meanings.

Notations	Meanings
S_i	The sensor node whose ID is i ($0 < i \leq N$)
N	Total number of sensor nodes in one MWSN
T^t	The t^{th} epoch
λ_i^t	Total number of target locations of S_i in T^t
$\text{Loc}_{i,j}^t$	The j^{th} target location of S_i during T^t
$\mu_{i,j}^t$	Total data item numbers of S_i generated at $\text{Loc}_{i,j}^t$ in T^t
$n_{i,j}^t$	Total number of qualified Top- k data items generated by S_i at $\text{Loc}_{i,j}^t$ in T^t
Q^t	A spatial-temporal Top- k query
R^t	The query result of Q^t
I_{Q^t}	The ID of Q^t
I_{MWSN}	The ID of an MWSN
$\text{QR}_{I_{\text{MWSN}}}$	The queried region in an MWSN whose ID is I_{MWSN}
Key_i	The pairwise key which is distributed to sensor node S_i
$\text{RT}_{S_i}^t$	The data report generated by S_i in T^t
$E_{\text{Key}_i}^{\{*\}}$	Symmetric encrypting operation with Key_i based on [34]
$E_{\text{OPE}}^{\{*\}}$	Encrypting operation based on the OPE encryption scheme [35]
$\text{RST}_{S_i}^t$	The processed result of $\text{RT}_{S_i}^t$
Ω_i	Total number of the queried locations encrypted in $\text{RST}_{S_i}^t$
$\gamma_{i,j}^t$	Total number of the sensed data items encrypted in $\text{DPP}_{i,j}^t$
R_{tpk}	Set of the qualified Top- k data items extracted from R^t

3.2. *Definitions.* In this section, we introduce the definitions of some terminologies used in this paper. Specifically, we define the terminologies used in this paper as follows:

- (i) Fine-grained spatial-temporal Top- k query: it is the query which tries to find out the top k sensed data items that have the biggest (or the smallest) scores among all the sensed data items generated in $\text{QR}_{I_{\text{MWSN}}}$ in T^t , where $\text{QR}_{I_{\text{MWSN}}}$ is a subregion of the deployment field of the MWSN whose ID is I_{MWSN} . The meta-language of a fine-grained spatial-temporal Top- k query Q^t in FMSCSs is shown in the following equation:

$$Q^t = \{I_{Q^t}, T^t, k, I_{\text{MWSN}}, \text{QR}_{I_{\text{MWSN}}}\}. \quad (1)$$

- (ii) Queried node and queried location: given a spatial-temporal Top- k query $Q^t = \{I_{Q^t}, T^t, k, I_{\text{MWSN}}, \text{QR}_{I_{\text{MWSN}}}\}$, if a target location of any mobile sensor node falls in $\text{QR}_{I_{\text{MWSN}}}$ in T^t , the target location is one of the queried locations of Q^t ; if at least one of the target locations of a mobile sensor node is one of the queried locations of Q^t , the sensor node is called a queried node of Q^t .
- (iii) Qualified Top- k data items: given a spatial-temporal Top- k query $Q^t = \{I_{Q^t}, T^t, k, I_{\text{MWSN}}, \text{QR}_{I_{\text{MWSN}}}\}$, if a sensed data item $D_{\text{qualified}}^t$ satisfies the following two conditions, it is called the qualified Top- k data item of Q^t : (1) $D_{\text{qualified}}^t$ was generated in $\text{QR}_{I_{\text{MWSN}}}$ and T^t ; (2) among all the sensed data items generated in T^t and T^t , there are at least $N_{Q^t} - k$ data items whose scores are smaller (or bigger) than the score of $D_{\text{qualified}}^t$, where N_{Q^t} refers to the total number of the sensed data items generated in $\text{QR}_{I_{\text{MWSN}}}$ and T^t .
- (iv) Data-proof Packet $\text{DPP}_{i,j}^t$: for any target location $\text{Loc}_{i,j}^t$ ($0 < j \leq \lambda_i^t$) of any mobile sensor node S_i

($1 \leq i \leq N$), Data-proof Packet $\text{DPP}_{i,j}^t$ refers to the subreport produced by S_i for the sensed data generated at $\text{Loc}_{i,j}^t$ during T^t . Specifically, $\text{DPP}_{i,j}^t$ consists of the pairwise-key-encrypted sensed data items and the OPE-encrypted scores (“OPE” is short for “order-preserving encryption” [35]) as well as some proof information generated by S_i at $\text{Loc}_{i,j}^t$ during T^t . More specific contents of $\text{DPP}_{i,j}^t$ will be described in Algorithm 1 in Section 4.

3.3. *Security Model.* In FMSCSs, fog nodes and the cloud servers are assumed to be untrusted, while most of the mobile sensor nodes and TA are trustworthy. We assume that the untrusted fog nodes and cloud servers are not only curious but also malicious. Specifically, a curious fog node or cloud server will try to disclose the sensed data items as well as the data scores computed based on the public scoring function, and a malicious fog node or cloud server will do its best to undermine the completeness of the results of the fine-grained spatial-temporal Top- k queries. To execute a malicious attack, an untrusted fog node may put none or only part of the qualified top k data items into the Top- k query result, and it may also put some fabricated data items and/or the unqualified-but-real ones into the query result when processing a spatial-temporal Top- k query. For example, suppose the complete query result should be $\{D_1^t, D_2^t, D_3^t\}$. Then, an incomplete query result may be $\{D_1^t\}$ or $\{D_1^t, D_4^t, D_{\text{fabricated}}^t\}$, where D_4^t is a real but unqualified sensed data item and $D_{\text{fabricated}}^t$ is a fabricated data item. An untrusted cloud server may also make some wrong deletions or replacements to undermine the integrity of the query results before it transmits the query results to end users.

In our security model, the privacy of the sensed data items, which are generated by the mobile sensor nodes in FMSCSs, and their corresponding scores should be

Ensure: target location set $\{\text{Loc}_{i,1}^t, \text{Loc}_{i,2}^t, \dots, \text{Loc}_{i,\lambda_i^t-1}^t, \text{Loc}_{i,\lambda_i^t}^t\}$; all the sensed data items generated by S_i in T^t ; the pairwise key Key_i ; the master key used for OPE;

Require: $\text{RT}_{S_i}^t$;

- (1) Compute the score of each sensed data item using the public scoring function;
- (2) **for** $j = 1$ to λ_i^t **do**
- (3) **if** $\mu_{i,j}^t = 0$ **then**
- (4) Set $\text{DPP}_{i,j}^t$ to $\{\text{Loc}_{i,j}^t, E_{\text{Key}_i}\{0, \text{Loc}_{i,j}^t\}\}$;
- (5) **end if**
- (6) **if** $\mu_{i,j}^t = 1$ **then**
- (7) Set $\text{DPP}_{i,j}^t$ to $\{\text{Loc}_{i,j}^t, E_{\text{Key}_i}\{1, \text{Loc}_{i,j}^t\}, E_{\text{Key}_i}\{d_{i,j,1}^t, \text{Loc}_{i,j}^t\}, E_{\text{OPE}}\{d_{i,j,1}^t\}, E_{\text{Key}_i}\{D_{i,j,1}^t, \text{Loc}_{i,j}^t\}\}$;
- (8) **end if**
- (9) **if** $\mu_{i,j}^t > 1$ **then**
- (10) Sort the sensed data items generated by S_i at $\text{Loc}_{i,j}^t$ in T^t according to their scores;
- (11) Set $\text{DPP}_{i,j}^t$ to $\{\text{Loc}_{i,j}^t, E_{\text{Key}_i}\{\mu_{i,j}^t, \text{Loc}_{i,j}^t\}, E_{\text{Key}_i}\{d_{i,j,1}^t, \text{Loc}_{i,j}^t\}, E_{\text{OPE}}\{d_{i,j,1}^t\}, E_{\text{Key}_i}\{1, D_{i,j,1}^t, \text{Loc}_{i,j}^t, \dots, E_{\text{OPE}}\{d_{i,j,\mu_{i,j}^t-1}^t\}\}, E_{\text{Key}_i}\{\mu_{i,j}^t - 1, D_{i,j,\mu_{i,j}^t-1}^t, \text{Loc}_{i,j}^t\}, E_{\text{OPE}}\{d_{i,j,\mu_{i,j}^t}^t\}, E_{\text{Key}_i}\{D_{i,j,\mu_{i,j}^t}^t, \text{Loc}_{i,j}^t\}\}$;
- (12) **end if**
- (13) **end for**
- (14) Set $\text{RT}_{S_i}^t$ to $\{i, t, E_{\text{Key}_i}\{\text{Loc}_{i,1}^t, \text{Loc}_{i,2}^t, \dots, \text{Loc}_{i,\lambda_i^t-1}^t, \text{Loc}_{i,\lambda_i^t}^t\}, \text{DPP}_{i,1}^t, \text{DPP}_{i,2}^t, \dots, \text{DPP}_{i,\lambda_i^t-1}^t, \text{DPP}_{i,\lambda_i^t}^t\}$;
- (15) Return $\text{RT}_{S_i}^t$.

ALGORITHM 1: Secure data preprocessing on S_i ($0 < i \leq N$).

protected. Other information, such as spatial-temporal Top- k query and the generation locations of the sensed data items, will be leaked to fog nodes. It is hard to enable fog nodes to process spatial-temporal Top- k query smoothly and successfully without such leaks. Fortunately, the leaked information brings little threat to the safety of the systems. Moreover, we assume each mobile sensor node is assumed to be equipped with the tamper-proof hardware, with the help of which the adversaries cannot disclose the encryption materials stored in the hardware even if they capture the sensor nodes [24].

3.4. Problem Statement and Design Goal. Under the system and the security models described above, the problem tackled in this paper can be presented as follows: how to make the end users in FMSCSs obtain the query results of the fine-grained spatial-temporal Top- k queries launched by them without disclosing the sensor data items and their corresponding scores to the fog nodes and the cloud servers and verify the completeness of the corresponding query result correctly and efficiently. Our design goal is to propose a novel scheme that enables efficient privacy-preservation and integrity-verifiable query processing for fine-grained spatial-temporal Top- k query in FMSCSs. Specifically, three objects as follows should be achieved:

- (i) The privacy preservation goal: our proposed scheme should preserve the privacy of the sensed data items and their scores collected from the mobile sensor nodes.
- (ii) The integrity verification goal: our proposed scheme should enable end users to verify the completeness of spatial-temporal Top- k query results, no matter what attacking means introduced in the security model are adopted.

- (iii) The efficiency goal: our proposed scheme should be effective in communication and computation. It should greatly decrease the additional communication cost of the sensor nodes, since the sensor nodes are energy-limited. Here, the additional communication cost mainly refers to the cost of transmitting the proof data that are used to verify the completeness of the query results.

4. Our Scheme STQ-SCS

This section presents our scheme STQ-SCS. We first make a high-level description of the scheme as follows. At first, each MWSNO obtains the secret keys from TA and preload the keys to its own MWSN. Then, using the secret keys, each sensor node encrypts its own sensed data items and the scores, and uploads the encrypted data items and their scores to the corresponding fog node. If an end user wants to retrieve the query result of a fine-grained spatial-temporal Top- k query, it sends the query to the cloud server or to the fog node directly if it is near the fog node of the target MWSN. If a cloud server receives the query, it first determines which fog node should be the target node of the query, and then sends the query to the target fog node. If the target fog node receives the query, it will work out all the qualified Top- k data items, put them into the query result packet, and send them to the cloud server or to the end user directly if the query is received by the fog node from the end user. If a cloud server receives the query result from the fog node, it will transmit the query result to the end user who is the launcher of the query.

As a whole, STQ-SCS can be mainly divided into five parts: (1) secret key distribution; (2) virtual-location construction; (3) secure data preprocessing; (4) secure spatial-temporal Top- k query processing; (5) completeness verification of the query results. In the following sections, the five parts of STQ-SCS are described in great detail.

4.1. Secret Key Distribution. In STQ-SCS, all secret keys used in FMSCSs are distributed by TA. To obtain the secret keys, each MWSNO sends a key-request message, which contains its own public key, the ID of its own MWSN, the IDs of the mobile sensor nodes in the MWSN, and some authentication information, to TA. After authenticating the identity of the MWSNO using some existing authentication method such as UAP-BCIoT [36], TA knows whether the MWSNO has the authority to obtain the secret keys or not. If TA determines to send the keys to the MWSNO, TA distributes a master key for the MWSN and a pairwise key for each mobile sensor node in the MWSN, encrypts them using the public key of the MWSNO, and then sends them to the MWSNO. The pairwise keys are generated based on the method in [34], while the master key is generated according to the scheme in [35]. Using the similar way, legal end users can also obtain the keys of each mobile sensor node in any MWSN from TA.

In our scheme, two encryption methods are leveraged to encrypt the sensed data items and their scores: one is the latest order preservation encryption (OPE) scheme [35] and the other one is the pairwise-key-based encryption [34]. The former is used to encrypt the scores of the sensed data items using the master keys, while the latter is used to encrypt the sensed data items and the proof data, such as the target locations of the sensor nodes and the ranking orders of the sensed data items, using the pairwise keys. Section 4.3 will describe this in detail.

4.2. Construction of the Virtual Grids. In STQ-SCS, the sensor deployment field is divided into many virtual grids. Each virtual grid should be as small as possible so that the central location of the grid can be approximately taken as the location of every point in the grid in real applications. Then, we design an ID distribution law for the virtual grids. Based on the law, the real locations of each mobile sensor node can be worked out easily if the IDs of the virtual grids where it has moved to are known.

Specifically, the ID distribution law is described as follows. Suppose the FMSCSs-deployed field is a $L * L$ square rectangle. STQ-SCS divides the rectangle into $\eta = (L/\zeta)^2$ small virtual grids, where ζ is a small digital number that can divide the length L with no remainder. Clearly, the smaller ζ is, the larger η is. Then, each virtual grid is given an ID, which is a sequence number ranging from 1 to η . The virtual grids in the first row at the upper side of the rectangle are given the IDs 1, 2, 3, ..., $L/(\zeta - 1)$, and L/ζ , respectively, from the left to the right in order; the IDs $L/(\zeta + 1)$, $L/(\zeta + 2)$, ..., $2 * (L/\zeta) - 1$, and $2 * (L/\zeta)$ are assigned to those in the second row orderly; ...; those in the last row have the IDs $\eta - L/(\zeta + 1)$, $\eta - L/(\zeta + 2)$, ..., $\eta - 1$, and η , respectively.

Using such an ID distribution law, each sensor node first works out the IDs of the virtual grid where it has moved to, and then takes the IDs as the coordinate values of its target locations.

4.3. Secure Data Preprocessing. This section describes how each sensor node generates its data report, which will be

uploaded to the corresponding fog node at the end of each epoch, based on its own sensed data items under the privacy-and-integrity preservation requirements. Specifically, for any sensor node S_i ($0 < i \leq N$), the procedure of data report generation in STQ-SCS is shown in Algorithm 1.

In the protocol, S_i firstly computes the score of each sensed data item generated by itself based on the public scoring function; then, it works out $DPP_{i,j}^t$ ($0 < j \leq \lambda_i^t$) for each of its target locations which it has been moved to during epoch T^t . To do this, three cases are considered: $\mu_{i,j}^t = 0$, $\mu_{i,j}^t = 1$, and $\mu_{i,j}^t > 1$. If $\mu_{i,j}^t = 0$, $DPP_{i,j}^t$ should include $E_{Key_i}\{0, Loc_{i,j}^t\}$ to show that no sensed data were generated by S_i at $Loc_{i,j}^t$ in epoch T^t , where $E_{Key_i}\{*\}$ is a symmetric encrypting operation with Key_i based on [34]; if $\mu_{i,j}^t = 1$, $DPP_{i,j}^t$ should contain $E_{Key_i}\{0, Loc_{i,j}^t\}$ to indicate that only one sensed data item was generated by S_i at $Loc_{i,j}^t$ in epoch T^t , and it also needs to include both the pairwise-key-encrypted score and the OPE-encrypted score of the only data item. The former will be used as part of the proof information for integrity verification, and the latter will be used by fog nodes to process spatial-temporal Top- k query smoothly. The only sensed data item should also be encoded using the pairwise key and included in $DPP_{i,j}^t$. If $\mu_{i,j}^t > 1$, the contents of $DPP_{i,j}^t$ are a little complex. Specifically, it contains not only the OPE-encrypted scores and the pairwise-key-encrypted data items and scores but also the chaining relationships of the ranked sensed data items. The chaining relationships, which are used to prevent the adversaries from destroying the integrity of the Top- k query results by dropping part of the qualified Top- k data items, are achieved by encrypting each sensed data item together with its ranking order number, which is called the sequence number in the following of this paper, using the pairwise key Key_i . Moreover, each sensed data item is bond together with its corresponding target location to further strengthen the integrity preservation of the Top- k query results. The final output $RT_{S_i}^t$ in Algorithm 1 is the very data report which will be uploaded to the corresponding fog node of S_i .

4.4. Secure Spatial-Temporal Top- k Query Processing. This section presents how a fine-grained spatial-temporal Top- k query is processed in FMSCSs in our proposed scheme STQ-SCS. When a cloud server receives a fine-grained spatial-temporal Top- k query from an end user, it first finds out the destination of the query according to the mapping relationships between the MWSN IDs and the fog nodes (Information about the mapping relationships is assumed to be stored in the cloud server). Then, the cloud server sends the query to the target fog node. When the target fog node receives the query, it processes the query according to Algorithm 2. After that, it sends the processing result back to the cloud server. If the query is sent from an end user, the fog node will send the query result back to the end user directly.

In Algorithm 2, the fog node first processes every data report uploaded by the sensor nodes in MWSN I_{MWSN} and then packets all the processing results of the data reports collected in the queried MWSN to form the final query result of the spatial-temporal Top- k query. Specifically, lines 1-9

```

Ensure:  $\{RT_{S_1}^t, RT_{S_2}^t, \dots, RT_{S_{N-1}}^t, RT_{S_N}^t\}; Q^t = \{I_{Q^t}, T^t, k, I_{MWSN}, QR_{I_{MWSN}}\};$ 
Require:  $R^t;$ 
(1) for  $i = 1$  to  $N$  do
(2)    $n[i] = 0;$ 
(3)   for  $j = 1$  to  $\lambda_i^t$  do
(4)     if  $Loc_{i,j}^t$  is in  $QR_{I_{MWSN}}$  then
(5)       put  $DPP_{i,j}^t$  into set  $\Theta;$ 
(6)        $n[i] = n[i] + 1;$ 
(7)     end if
(8)   end for
(9) end for
(10) Find out the pairwise-key-encrypted qualified Top- $k$  data items among all the pairwise-key-encrypted data items in set  $\Theta$  according to their corresponding OPE-encrypted scores;
(11) Calculate  $n_{i,j}^t$  for each  $i \in [1, N]$  and  $j \in [1, \lambda_i^t];$ 
(12) for  $i = 1$  to  $N$  do
(13)   if  $n[i] = 0$  then
(14)     Set  $RST_{S_i}^t$  to  $\{i, t, E_{Key_{i,t}}\{Loc_{i,1}^t, Loc_{i,2}^t, \dots, Loc_{i,\lambda_i^t}^t\}\}$ 
(15)   else
(16)     for  $j = 1$  to  $n[i]$  do
(17)       if  $\mu_{i,x_j}^t = 0$  then
(18)         Set  $DPP_{i,x_j}^t$  to  $\{E_{Key_i}\{0, Loc_{i,x_j}^t\}\};$ 
(19)       end if
(20)       if  $n_{i,x_j}^t = 0, \mu_{i,x_j}^t > 0$  then
(21)         set  $DPP_{i,x_j}^t$  to  $\{E_{Key_i}\{d_{i,x_j,1}^t, Loc_{i,x_j}^t\}\};$ 
(22)       end if
(23)       if  $0 < n_{i,x_j}^t = \mu_{i,x_j}^t \leq k$  then
(24)         if  $n_{i,x_j}^t = 1$  then
(25)           Set  $DPP_{i,x_j}^t$  to  $\{E_{Key_i}\{1, Loc_{i,x_j}^t\}, E_{Key_i}\{D_{i,x_j,1}^t, Loc_{i,x_j}^t\}\};$ 
(26)         end if
(27)         if  $n_{i,x_j}^t > 1$  then
(28)           set  $DPP_{i,x_j}^t$  to  $\{n_{i,x_j}^t, E_{Key_i}\{\mu_{i,x_j}^t, Loc_{i,x_j}^t\}, E_{Key_i}\{1, D_{i,x_j,1}^t, Loc_{i,x_j}^t\}, \dots, E_{Key_i}\{\mu_{i,x_j}^t - 1, D_{i,x_j,\mu_{i,x_j}^t-1}^t, Loc_{i,x_j}^t\}, E_{Key_i}\{D_{i,x_j,\mu_{i,x_j}^t}^t, Loc_{i,x_j}^t\}\};$ 
(29)         end if
(30)       end if
(31)       if  $0 < n_{i,x_j}^t \leq k, \mu_{i,x_j}^t > n_{i,x_j}^t$  then
(32)         if  $\mu_{i,x_j}^t = n_{i,x_j}^t + 1$  then
(33)           Set  $DPP_{i,x_j}^t$  to  $\{n_{i,x_j}^t, E_{Key_i}\{\mu_{i,x_j}^t, Loc_{i,x_j}^t\}, E_{Key_i}\{1, D_{i,x_j,1}^t, Loc_{i,x_j}^t\}, \dots, E_{Key_i}\{n_{i,x_j}^t, D_{i,x_j,n_{i,x_j}^t}^t, Loc_{i,x_j}^t\}, E_{Key_i}\{D_{i,x_j,\mu_{i,x_j}^t}^t, Loc_{i,x_j}^t\}\};$ 
(34)         end if
(35)         if  $\mu_{i,x_j}^t > n_{i,x_j}^t + 1$  then
(36)           set  $DPP_{i,x_j}^t$  to  $\{n_{i,x_j}^t, E_{Key_i}\{\mu_{i,x_j}^t, Loc_{i,x_j}^t\}, E_{Key_i}\{1, D_{i,x_j,1}^t, Loc_{i,x_j}^t\}, \dots, E_{Key_i}\{n_{i,x_j}^t, D_{i,x_j,n_{i,x_j}^t}^t, Loc_{i,x_j}^t\}, E_{Key_i}\{n_{i,x_j}^t + 1, D_{i,x_j,n_{i,x_j}^t+1}^t, Loc_{i,x_j}^t\}\};$ 
(37)         end if
(38)       end if
(39)     end for
(40)   Set  $RST_{S_i}^t$  to  $\{i, t, E_{Key_i}\{Loc_{i,1}^t, Loc_{i,2}^t, \dots, Loc_{i,\lambda_i^t}^t, DPP_{i,x_1}^t, DPP_{i,x_2}^t, \dots, DPP_{i,x_{n[i]}}^t, DPP_{i,x_{n[i]}}^t\};$ 
(41)   end if
(42) end for
(43) Return set  $\{I_{Q^t}, RST_{S_1}^t, RST_{S_2}^t, \dots, RST_{S_{N-1}}^t, RST_{S_N}^t\}.$ 

```

ALGORITHM 2: Secure spatial-temporal Top- k query processing on the target fog node.

aim to find out the number of locations that fall in $QR_{I_{MWSN}}$ of each sensor node in MWSN I_{MWSN} and the corresponding Data – proofPackets generated at those locations; from lines 12 to 42, there is a big “for” loop, which is used to process every report generated in MWSN I_{MWSN} in T^t . Line 14 shows

the processing result of $RT_{S_i}^t$ considering the case that no target location of S_i falls in $QR_{I_{MWSN}}$ in T^t ; lines 16–39 describe the procedure of processing $RT_{S_i}^t$ considering the case that there is at least one location of S_i that falls in $QR_{I_{MWSN}}$ in T^t . In the abovementioned latter case, all the Data – proofPackets

that correspond to the target locations located in $QR_{I_{MWSN}}$ are processed based on the exact values of μ_{i,x_j}^t and/or n_{i,x_j}^t , where μ_{i,x_j}^t and n_{i,x_j}^t denote the total data number and the qualified data number, respectively, corresponding to the location Loc_{i,x_j}^t , which is supposed to be in the queried region $QR_{I_{MWSN}}$. During the procedure of processing the Data – proofPackets, the OPE-encrypted items are all removed from the original Data – proofPackets since the only use of them is to make fog nodes find out the qualified Top- k data items encrypted with the pairwise keys. Moreover, all the unqualified data items except for the one which follows the last qualified Top- k data item in each Data – proofPackets are also removed from each original Data – proofPackets, and the reserved one will be used for completeness verification of the spatial-temporal Top- k query results.

4.5. Completeness Verification of the Query Results. The procedure for an end user to verify the completeness of the Top- k query result R^t is presented in Algorithm 3, the output of which is the value of the Boolean variable completeness. If completeness is false, R^t is considered as incomplete; otherwise, R^t is complete and the final R_{tpk} in Algorithm 3 is composed of all the qualified Top- k data items corresponding to the fine-grained spatial-temporal Top- k query Q^t .

The main idea of Algorithm 3 to verify the completeness of R^t is to find out the minimal data score of the qualified Top- k data items and the maximal score of the unqualified ones generated in the queried region from R^t , and compare them with each other. Normally, the former should be bigger than the latter if the query aims to find out the biggest top k data items. If this condition does not hold in R^t , R^t is considered incomplete. However, it is not correct yet to declare that R^t is complete even if such a condition holds in R^t . Before doing such a comparison, it is necessary to check whether each sensor report was processed properly by the compromised fog node (lines 2–53 in Algorithm 3) based on the proof information included in R^t . To achieve this, each Data – proofPacket in R^t should be checked. When checking the Data – proofPackets, three cases need to be considered, namely, $\gamma_{i,x_j}^t = 0$ (lines 16–25), $\gamma_{i,x_j}^t = 1$ (lines 26–32), and $\gamma_{i,x_j}^t > 1$ (lines 33–51). If $\gamma_{i,x_j}^t = 0$, either S_i did not generate any data items at Loc_{i,x_j}^t in T^t or no data item generated by S_i at Loc_{i,x_j}^t in T^t is the qualified Top- k data item. Thus, in such a case, either $E_{Key_i} \left\{ d_{i,x_j,1}^t, Loc_{i,x_j}^t \right\}$ or $E_{Key_i} \left\{ 0, Loc_{i,x_j}^t \right\}$ should be originally included in DPP_{i,x_j}^t in R^t . If $\gamma_{i,x_j}^t = 1$, the data item included in DPP_{i,x_j}^t should be a qualified Top- k data item according to lines 24–26 in Algorithm 2. If $\gamma_{i,x_j}^t > 1$, according to lines 27–38 in Algorithm 2, the fog node must have made some illegal query-processing operations if any of the following cases happens (lines 33–35 in Algorithm 3): (a) n_{i,x_j}^t is not included in DPP_{i,x_j}^t in R^t ; (b) no sensed data item in DPP_{i,x_j}^t is encrypted with a sequence number; (c) the sequence numbers encrypted in DPP_{i,x_j}^t are not sorted in ascending order from 1; (d) any sensed data item encrypted in DPP_{i,x_j}^t is not originally encrypted with Loc_{i,x_j}^t ; and (e) $E_{Key_i} \left\{ \mu_{i,x_j}^t, Loc_{i,x_j}^t \right\}$ is not originally included in DPP_{i,x_j}^t .

Moreover, in the case that $\gamma_{i,x_j}^t > 1$, γ_{i,x_j}^t should be equal to either n_{i,x_j}^t or $n_{i,x_j}^t + 1$ according to lines 27–38 in Algorithm 2 where n_{i,x_j}^t is included in R^t . Thus, in lines 36–50 in Algorithm 3, the abovementioned two cases are considered, respectively, to detect the integrity of R^t .

5. Security Analysis

5.1. Analysis of STQ-SCS on Privacy Preservation

Theorem 1. *Our scheme STQ-SCS is able to preserve the privacy of both the sensed data items and its scores for fine-grained spatial-temporal Top- k query in FMSCSs under the security model presented in this paper.*

Proof. According to Algorithm 1, before being uploaded to fog nodes, all sensed data items are encrypted with the pairwise keys and all the data scores are encrypted with the master keys [35] by the sensor nodes in FMSCSs. Meanwhile, all the encryption keys should only be obtained from TA after authentication according to the key-distribution method used in STQ-SCS, and the fog nodes and the cloud servers are not able to obtain the keys and thus cannot disclose the values of the sensed data items and their scores. Since the cloud servers and the fog nodes are assumed to be curious and/or malicious while other parties in FMSCSs are assumed to be trustworthy in our security model, the privacy of the sensed data items and their scores can be preserved for fine-grained spatial-temporal Top- k query in FMSCSs using our scheme STQ-SCS. \square

5.2. Analysis of STQ-SCS on Completeness Verification

Theorem 2. *Suppose a queried node S_i ($\forall i \in [1, N]$) generated $\mu_{i,j}^t$ ($\mu_{i,j}^t > 0$) data items at a queried location $Loc_{i,j}^t$ ($\forall j \in [1, \lambda_i^t]$) in epoch T^t , where there are $n_{i,j}^t$ ($0 < n_{i,j}^t \leq k$) qualified Top- k data items. If at least one of those qualified Top- k data items is dropped from $DPP_{i,j}^t$ ($\forall i \in [1, N], \forall j \in [1, \lambda_i^t]$) in the query result R^t of $Q^t = \{I_Q, T^t, k, I_{MWSN}, QR_{I_{MWSN}}\}$ by the fog node or the cloud server which generates and/or transmits R^t , the incomplete R^t must be detected by end users with a 100% successful rate based on our scheme STQ-SCS.*

Proof. Since the fog node or the cloud server does not know Key_i , if it inserts the sensed data items that are encrypted with some other keys rather than Key_i into $DPP_{i,j}^t$ ($\forall i \in [1, N], \forall j \in [1, \lambda_i^t]$), the incomplete R^t must be detected by the end user according to lines 6–9 in Algorithm 3. Moreover, according to lines 33–35 in Algorithm 3, R^t must be also considered as incomplete if the fog node or the cloud server puts any encrypted data item, which was generated by S_i in T^t at some other location rather than $Loc_{i,j}^t$, into $DPP_{i,j}^t$. Thus, in the following of this proof, we need only to consider the situation that all the encrypted sensed data items left in $DPP_{i,j}^t$ after being processed by the fog node are the real ones which were generated by S_i ($\forall i \in [1, N]$) at $Loc_{i,j}^t$ in T^t (but some or all of them may not be the qualified ones). Then, if at least one qualified sensed data items generated by S_i at $Loc_{i,j}^t$ in T^t is discarded by the fog node or the cloud server, one of

Ensure: $R_t = \{I_{Q^t}, RST_{S_1}^t, RST_{S_2}^t, \dots, RST_{S_{N-1}}^t, RST_{S_N}^t\}$; $Q^t = \{I_{Q^t}, T^t, k, I_{MWSN}, QR_{I_{MWSN}}\}$; $\{Key_1^t, Key_2^t, \dots, Key_{N-1}^t, Key_N^t\}$.
Require: Completeness.

- (1) $R_{tpk} = \emptyset$; $V_{nonTop} = \emptyset$; Completeness = true;
- (2) **for** $i = 1$ to N **do**
- (3) **if** $(RST_{S_i}^t \notin R_t) \parallel (RST_{S_i}^t \text{ contains no pairwise-key-encrypted target locations})$ **then**
- (4) Set Completeness = false; return Completeness;
- (5) **end if**
- (6) Decrypt all the ciphertext in $RST_{S_i}^t$ with Key_i ;
- (7) **if** The end user cannot decrypt the ciphertext normally **then**
- (8) Completeness = false; return Completeness;
- (9) **end if**
- (10) Calculate the value of Ω_i which is the total number of the queried locations in $RST_{S_i}^t$;
- (11) **for** $j = 1$ to Ω_i **do**
- (12) **if** DPP_{i,x_j}^t is not originally in $RST_{S_i}^t$ (DPP_{i,x_j}^t is a Data-proof Packet corresponding to Loc_{i,x_j}^t which is in $QR_{I_{MWSN}}$) **then**
- (13) Completeness = false; return Completeness;
- (14) **end if**
- (15) Calculate the value of γ_{i,x_j}^t which is the total number of the sensed data items in DPP_{i,x_j}^t ;
- (16) **if** $\gamma_{i,x_j}^t = 0$ **then**
- (17) **if** $E_{Key_i}\{d_{i,x_j,1}^t, Loc_{i,x_j}^t\}$ is originally in DPP_{i,x_j}^t in R^t **then**
- (18) $V_{nonTop} = V_{nonTop} \cup \{d_{i,x_j,1}^t\}$;
- (19) Continue;
- (20) **else if** $E_{Key_i}\{0, Loc_{i,x_j}^t\}$ is originally in DPP_{i,x_j}^t in R^t **then**
- (21) Continue;
- (22) **else**
- (23) Completeness = false; return Completeness;
- (24) **end if**
- (25) **end if**
- (26) **if** $\gamma_{i,x_j}^t = 1$ **then**
- (27) **if** $DPP_{i,x_j}^t \neq \{E_{Key_i}\{1, Loc_{i,x_j}^t\}, E_{Key_i}\{D_{i,x_j,1}^t, Loc_{i,x_j}^t\}\}$ **then**
- (28) Completeness = false; return Completeness;
- (29) **end if**
- (30) $R_{tpk} = R_{tpk} \cup \{D_{i,x_j,1}^t\}$;
- (31) Continue;
- (32) **end if**
- (33) **if** (n_{i,x_j}^t is not included in DPP_{i,x_j}^t in R^t) \parallel (no sensed data item in DPP_{i,x_j}^t is encrypted with a sequence number) \parallel (the sequence numbers encrypted in DPP_{i,x_j}^t are not sorted in ascending order from 1) \parallel (any sensed data item encrypted in DPP_{i,x_j}^t is not originally encrypted with Loc_{i,x_j}^t) \parallel ($E_{Key_i}\{\mu_{i,x_j}^t, Loc_{i,x_j}^t\}$ is not originally included in DPP_{i,x_j}^t) **then**
- (34) Completeness = false; return Completeness;
- (35) **end if**
- (36) **if** $n_{i,x_j}^t = \gamma_{i,x_j}^t$ **then**
- (37) **if** $\gamma_{i,x_j}^t \neq \mu_{i,x_j}^t$ **then**
- (38) Completeness = false; return Completeness;
- (39) **else**
- (40) $R_{tpk} = R_{tpk} \cup \{D_{i,x_j,1}^t, D_{i,x_j,2}^t, \dots, D_{i,x_j,\gamma_{i,x_j}^t}^t\}$;
- (41) **end if**
- (42) **else if** $n_{i,x_j}^t = \gamma_{i,x_j}^t - 1$ **then**
- (43) **if** ($E_{Key_i}\{D_{i,x_j,\mu_{i,x_j}^t}^t, Loc_{i,x_j}^t\}$ is included in DPP_{i,x_j}^t) $\&\&$ ($\gamma_{i,x_j}^t \neq \mu_{i,x_j}^t$) **then**
- (44) Completeness = false; return Completeness;
- (45) **end if**
- (46) $R_{tpk} = R_{tpk} \cup \{D_{i,x_j,1}^t, D_{i,x_j,2}^t, \dots, D_{i,x_j,\mu_{i,x_j}^t}^t\}$;
- (47) $V_{nonTop} = V_{nonTop} \cup \{f(D_{i,x_j,\gamma_{i,x_j}^t}^t)\}$;
- (48) **else**
- (49) Completeness = false; return Completeness;

```

(50)   end if
(51)   end for
(52)   end for
(53)   if (( $V_{\text{nonTop}} = \emptyset$ ) || ( $\text{SIZE}(R_{tpk}) \neq k$ )) then
(54)     Completeness = false; return Completeness;
(55)   end if
(56)   if  $f(\text{MIN}(R_{tpk})) < \text{MAX}(V_{\text{nonTop}})$  then
(57)     Completeness = false; return Completeness;
(58)   end if
(59)   Return Completeness.

```

ALGORITHM 3: Completeness verification of the query result R^t .

the following two cases must appear: (1) the fog node or the cloud server has dropped all the sensed data items from $\text{DPP}_{i,j}^t$ when producing or transmitting R^t and (2) the fog node or the cloud server has just discarded only a part of the sensed data items from $\text{DPP}_{i,j}^t$, and the discarded data items contain some qualified one/ones.

First of all, consider the case that the fog node or the cloud server has deleted all the sensed data items from $\text{DPP}_{i,j}^t$. In this case, the fog node or the cloud server should leave $E_{\text{Key}_i}\{d_{i,j,1}^t, \text{Loc}_{i,j}^t\}$ in $\text{DPP}_{i,j}^t$ in $\text{RST}_{S_i}^t$ of R^t to avoid being detected according to lines 16–25 in Algorithm 3 because it cannot generate the legal encryption item $E_{\text{Key}_i}\{0, \text{Loc}_{i,j}^t\}$. Then, $d_{i,j,1}^t$ should be put into V_{nonTop} according to lines 17–18 in Algorithm 3, and some real but unqualified sensed data items generated in $\text{QR}_{I_{\text{MWSN}}}^t$ and T^t must be put into R_{tpk} to make the number of the elements in R_{tpk} equal to k according to lines 53–55 in Algorithm 3. If the discarded sensed data items contain some qualified one/ones, $d_{i,j,1}^t$ must be the score of a qualified Top- k data item. Then, $f(\text{MIN}(R_{tpk}))$ must be smaller than $\text{MAX}(V_{\text{nonTop}})$ because the score of any qualified Top- k data item must be bigger than that of any real but unqualified one generated in $\text{QR}_{I_{\text{MWSN}}}^t$ and T^t assuming all data scores are distinct. Thus, according to lines 56–58 in Algorithm 3, the incomplete R^t must be detected by the end user.

Then, consider the case that the fog node or the cloud server has just deleted a part of the sensed data items from $\text{DPP}_{i,j}^t$, and the deleted data items contain some qualified one/ones. In this case, two situations should be discussed. One is that all the sensed data items encrypted with sequence order numbers are deleted from, while the other is that at least one sensed data item encrypted with a sequence number is left in $\text{DPP}_{i,j}^t$ after being processed. In the first situation, $E_{\text{Key}_i}\{D_{i,j,\mu_{i,j}^t}^t, \text{Loc}_{i,j}^t\}$ must be left in $\text{DPP}_{i,j}^t$ after being processed, and there must be $\text{DPP}_{i,x_j}^t \neq \{E_{\text{Key}_i}\{1, \text{Loc}_{i,j}^t\}, E_{\text{Key}_i}\{D_{i,j,1}^t, \text{Loc}_{i,j}^t\}\}$ since $\mu_{i,j}^t \neq 1$ in this situation and $E_{\text{Key}_i}\{1, \text{Loc}_{i,j}^t\}$ must not be included in $\text{DPP}_{i,j}^t$. According to lines 26–29 in Algorithm 3, the incomplete R^t must be detected by the end user. Then, consider the second situation. To make the sequence numbers encrypted with the sensed data items in $\text{DPP}_{i,j}^t$ in $\text{RST}_{S_i}^t$ of R^t ascends from 1 orderly (Lines 33–35 in Algorithm 3), the fog node or the cloud server must delete all the sensed data items in one of

the sets $\Phi_1, \Phi_2, \Phi_3, \Phi_4$, and Φ_5 from $\text{DPP}_{i,j}^t$. The five sets are shown in equation (2), where $1 < \omega < \mu_{i,j}^t - 1$.

$$\begin{cases}
\Phi_1 = \{D_{i,j,\omega}^t, D_{i,j,\omega+1}^t, \dots, D_{i,j,\mu_{i,j}^t-1}^t\}, \\
\Phi_2 = \{D_{i,j,\mu_{i,j}^t-1}^t\}, \\
\Phi_3 = \{D_{i,j,\omega}^t, D_{i,j,\omega+1}^t, \dots, D_{i,j,\mu_{i,j}^t}^t\}, \\
\Phi_4 = \{D_{i,j,\mu_{i,j}^t-1}^t, D_{i,j,\mu_{i,j}^t}^t\}, \\
\Phi_5 = \{D_{i,j,\mu_{i,j}^t}^t\}.
\end{cases} \quad (2)$$

If the fog node or the cloud server discards the sensed data items/item in set Φ_1 or Φ_2 from $\text{DPP}_{i,j}^t$ when processing $\text{DPP}_{i,j}^t$, $E_{\text{Key}_i}\{D_{i,j,\mu_{i,j}^t}^t, \text{Loc}_{i,j}^t\}$ and $E_{\text{Key}_i}\{1, D_{i,j,1}^t, \text{Loc}_{i,j}^t\}$ must be left in $\text{DPP}_{i,j}^t$ after being processed, which means that $\gamma_{i,j}^t$ is bigger than 1. According to lines 36–50 in Algorithm 3, the fog node has to either set $n_{i,j}^t$ to $\gamma_{i,j}^t$ or $\gamma_{i,j}^t - 1$ in $\text{DPP}_{i,j}^t$ in $\text{RST}_{S_i}^t$ of R^t to prevent the incomplete R^t from being detected. Even though, the incomplete R^t must also be detected by the end user according to lines 36–38 and 42–45 in Algorithm 3 because $\gamma_{i,j}^t$ must not be equal to $\mu_{i,j}^t$ in this case and $E_{\text{Key}_i}\{D_{i,x_j,\mu_{i,j}^t}^t, \text{Loc}_{i,j}^t\}$ is included in $\text{DPP}_{i,j}^t$ at the same time.

If the fog node or the cloud server deletes the sensed data items/item in set Φ_3, Φ_4 , or Φ_5 from $\text{DPP}_{i,j}^t$, the encryption item $E_{\text{Key}_i}\{\gamma_{i,j}^t, D_{i,j,\gamma_{i,j}^t}^t, \text{Loc}_{i,j}^t\}$ should be left in $\text{DPP}_{i,j}^t$. Then, if $\gamma_{i,j}^t = 1$, the incomplete R^t must be detected by the end user according to lines 26–29; if $\gamma_{i,j}^t > 1$, since $\gamma_{i,j}^t \neq \mu_{i,j}^t$ in this case, the fog node or the cloud server has to set $n_{i,j}^t$ to $\gamma_{i,j}^t - 1$ in $\text{DPP}_{i,j}^t$ in $\text{RST}_{S_i}^t$ of R^t to make the incomplete R^t free from being detected according to lines 36–50 in Algorithm 3. Then, $f(D_{i,j,\gamma_{i,j}^t}^t)$ will be put into set V_{nonTop} according to lines 42–47 in Algorithm 3. Because some dropped sensed data item/items is/are qualified Top- k data item/items, $D_{i,j,\gamma_{i,j}^t}^t$ must also be a qualified Top- k data item. Since the number of the sensed data items in R_{tpk} should be k , some real but unqualified Top- k data items whose scores are smaller than $f(D_{i,j,\gamma_{i,j}^t}^t)$ must be put into set R_{tpk} . Thus, there

must be $f(\text{MIN}(R_{tpk})) < \text{MAX}(V_{\text{nonTop}})$, and the incomplete R^t must be detected by the end user according to lines 56–58 in Algorithm 3.

Thus, if the fog node drops at least one qualified sensed data items from $\text{DPP}_{i,j}^t$, the end user in FMSCSs is able to detect the incomplete R^t with a successful rate of 100% based on STQ-SCS, and Theorem 2 holds. \square

Theorem 3. *Under the security model presented in this paper, any end user in FMSCSs can detect the incomplete query results of fine-grained spatial-temporal Top-k queries with a 100% successful rate based on our scheme STQ-SCS.*

Proof. According to the security model, untrusted parties (the fog nodes and the cloud servers) cannot fabricate the pairwise-key-encrypted sensed data items, which cannot be detected by end users, because the untrusted parties cannot obtain the legal pairwise keys. Thus, for any fine-grained spatial-temporal Top-k query Q^t , if its query result R^t is incomplete, at least one qualified sensed data item must be discarded by the fog node or the cloud server when producing and/or transmitting R^t . In other words, there must be at least one queried sensor node S_i ($\forall i \in [1, N]$) whose corresponding Data – proofPacket $\text{DPP}_{i,j}^t$ at location $\text{Loc}_{i,j}^t$ ($\forall j \in [1, \lambda_i^t]$) satisfies the following condition: at least one qualified sensed data item was deleted from $\text{DPP}_{i,j}^t$ by the fog node or the cloud server when producing and/or transmitting R^t . Then, according to Theorem 2, the incomplete R^t must be detected by the end user in FMSCSs based on our scheme STQ-SCS. Thus, Theorem 3 holds. \square

6. Computation Complexity Analysis

This section analyzes the computation complexity of the three schemes presented above.

Firstly, the computation complexity of Algorithm 1 is analyzed as follows. Since most of the statements in Algorithm 1 are the loop body of the “for” loop statements in Algorithm 1, the computation complexity of Algorithm 1 should be that the loop numbers multiply the computation complexity of the loop body. In the loop body, there are only three conditional statements. Thus, the computation complexity of the loop body depends on the pairwise-key encryption methods used in STQ-SCS and the total length of the data that need to be encrypted as well as the computation complexity of OPE. Although different pairwise-key cryptography methods, such as [34, 37], may have different computation complexities, they are considered lightweight generally and fit for the resource-limited sensor nodes [38, 39], let alone the fog nodes which are much more powerful than the sensor nodes. Moreover, OPE also has low computation complexity according to [35]. For each $\text{DPP}_{i,j}^t$ ($0 < i \leq N$, $0 < j \leq \mu_{i,j}^t$), the length of the data that need to be encrypted varies according to $\mu_{i,j}^t$, which symbolizes the total number of the sensed data items generated by S_i at $\text{Loc}_{i,j}^t$ in T^t . Let l_D and l_d denote the bit length of a sensed data item and that of a data score, respectively, l_n symbolizes not only the bit length of a sequence number but also that of $\mu_{i,j}^t$, l_{Loc} refers to the bit length of a virtual location, and $l_{i,j}^{\text{OPE}}$

TABLE 2: Default parameter settings.

Parameters	Default value
N	300
T (length of each epoch)	100 s
T_{mobile} (period for a sensor node to keep moving)	5 s
T_{static} (period for a sensor node to keep static)	5 s
m_{speed} (moving speed of each mobile sensor node)	5 m/s
r_{mobile} (ratio of the mobile sensor nodes to the total ones)	100%
$\text{MWSN}_{\text{size}}$ (size of the deployment field of each MWSN)	$400 \times 400 \text{ m}^2$
R (communication radius of each sensor node)	50 m
r_D (data generation rate of each sensor node)	2 item(s)
q_{period} (period for the end user to launch a query)	5 s
q_{radius} (radius of the queried region which is a circle)	50 m
l_D (length of a sensed data item)	400 bits
l_d (length of a data score)	20 bits
l_n (length of a sequence number)	10 bits
l_{id} (length of an ID number)	10 bits
l_t (length of a time data)	32 bits
l_{Loc} (length of each two-dimensional location)	128 bits
l_{VLoc} (length of each virtual location)	16 bits
e_{send} (cost of sending one bit data)	1 mJ
e_{receive} (cost of receiving one bit data)	1 mJ

and $l_{i,j}^{\text{PW}}$ denote the bit length of the data that need to be encrypted using OPE and that of those encoded adopting the pairwise-key encryption method, respectively, in $\text{DPP}_{i,j}^t$. Then, the values of $l_{i,j}^{\text{OPE}}$ and $l_{i,j}^{\text{PW}}$ can be worked out using equations (3) and (4), respectively, according to Algorithm 1.

$$l_{i,j}^{\text{OPE}} = \begin{cases} 0, & \text{if } \mu_{i,j}^t = 0, \\ l_d, & \text{if } \mu_{i,j}^t = 1, \\ \mu_{i,j}^t \times l_d, & \text{if } \mu_{i,j}^t \geq 2, \end{cases} \quad (3)$$

$$l_{i,j}^{\text{PW}} = \begin{cases} l_n + l_{\text{Loc}}, & \text{if } \mu_{i,j}^t = 0, \\ l_n + l_d + l_D + 3l_{\text{Loc}}, & \text{if } \mu_{i,j}^t = 1, \\ (l_n + l_D + l_{\text{Loc}})\mu_{i,j}^t + l_d + 2l_{\text{Loc}}, & \text{if } \mu_{i,j}^t \geq 2. \end{cases} \quad (4)$$

Secondly, pay attention to Algorithm 2. The computation complexity of lines 1–9 is $O(\sum_{i=1}^N \lambda_i^t)$; the computation complexity of line 10 depends on the adopted sorting algorithm and the total number of sensed data items generated in T^t and QR_{MWSN} ; that of line 11 is $O(\sum_{i=1}^N \lambda_i^t)$; that of lines 12–43 in Algorithm 2 is $O(N)$ in the best case (e.g., $n[i]$ is always 0 for each $i \in [1, N]$) and is $O(\sum_{i=1}^N n[i])$ in the worst case (e.g., $n[i]$ is not equal to 0 for each $i \in [1, N]$).

Finally, it is the turn of Algorithm 3, which mainly consists of one outer “for” loop whose loop body contains an inner “for” loop. In the loop body of the outer loop, the computation complexity of line 6 is the highest among all the statements that are in the loop body of the outer loop and out of the inner loop. If decrypting one encryption item $E_{\text{Key}_i}\{*\}$ is taken as one operation, the operation number of line 6

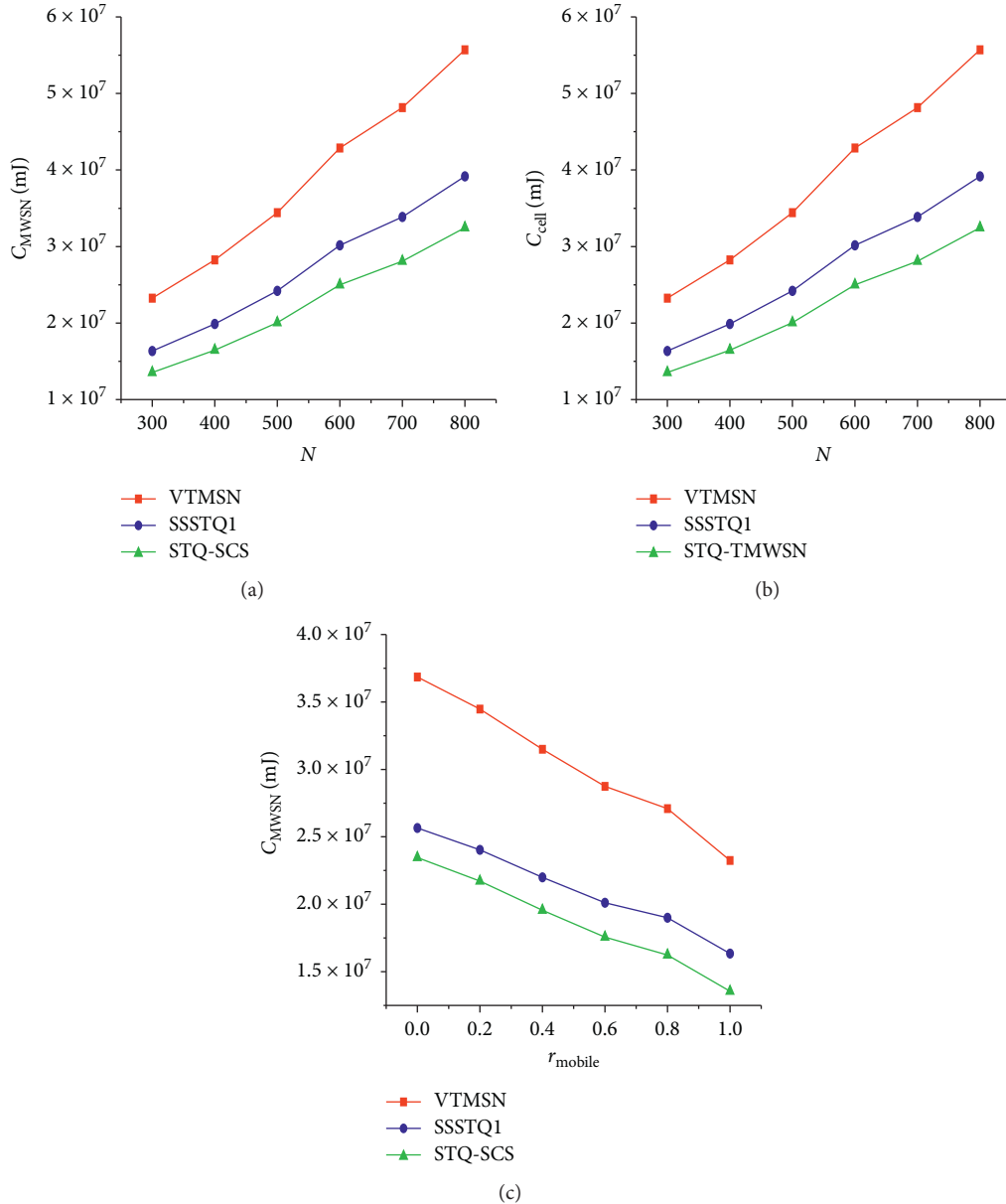


FIGURE 3: C_{MWSN} with different settings of r_D (a), N (b), and r_{mobile} (c).

should be $n[i] + 1$ according to line 40 in Algorithm 2. Then, the computation complexity of Algorithm 3 should be $O(\sum_{i=1}^N (n[i] + 1 + \Omega_i))$.

7. Performance Evaluation

In this section, we evaluate the performances of our proposed scheme STQ-SCS through extensive simulations taking OMNET++ as the simulation tool.

7.1. Metrics and Experimental Setup. The performance of STQ-SCS on energy efficiency is evaluated mainly by testing the additional communication cost, which is brought by

transmitting the proof data, because other data such as the sensed data items always need to be transmitted no matter what kind of methods are used to ensure the security of the query. Specifically, the metrics used in our simulations are listed as follows.

- (i) Additional communication cost in an MWSN (C_{MWSN}): total energy consumed by transmitting all the proof data produced in an MWSN and an epoch to the fog node in the MWSN. Since the sensor nodes are energy-limited, the additional energy cost brought by transmitting the proof data from each MWSN to its corresponding fog node should be given more attention to.

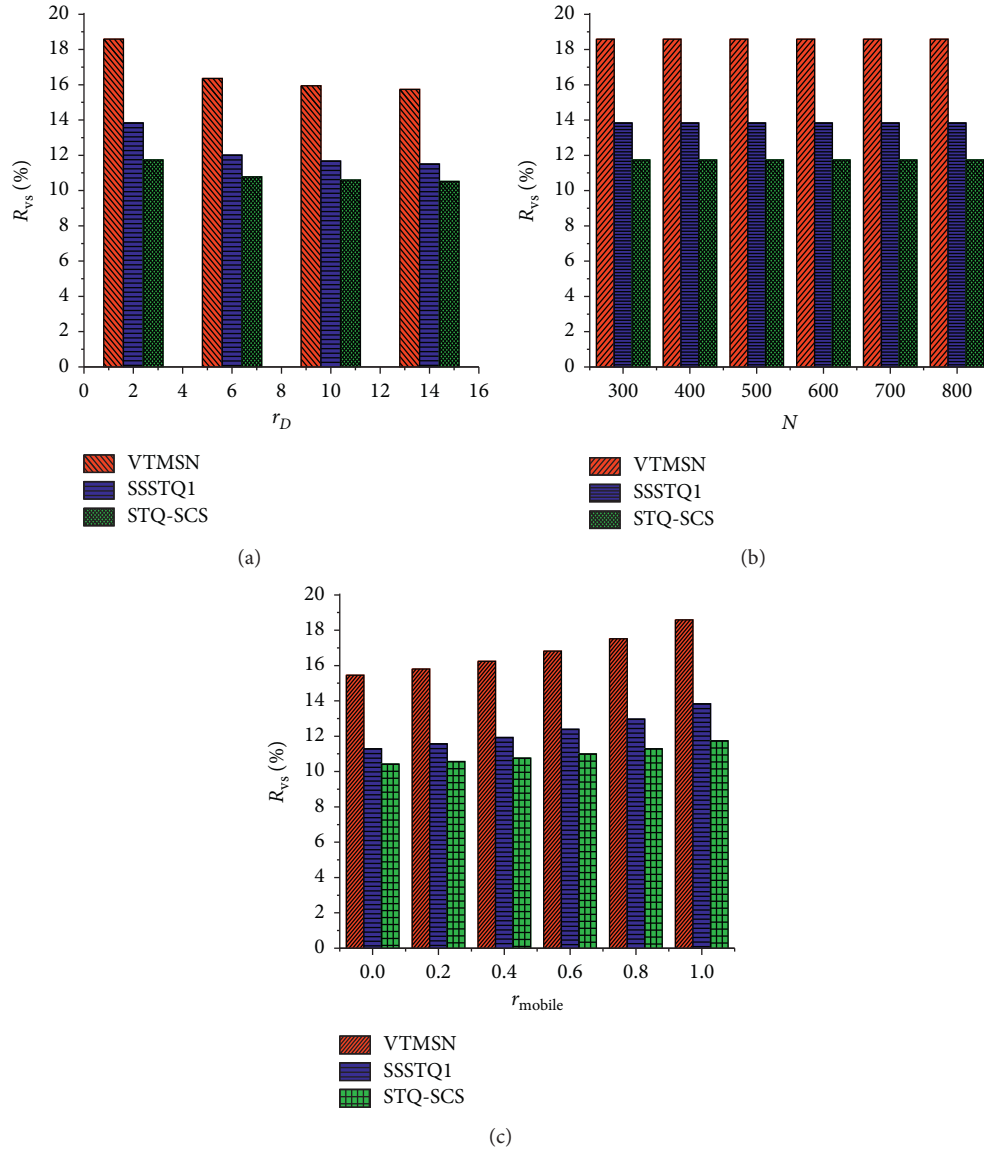


FIGURE 4: R_{vs} with different settings of r_D (a), N (b), and r_{mobile} (c).

- (ii) Proof-data ratio (R_{vs}): the ratio of C_{MWSN} to $C_{reports}$. Here, $C_{reports}$ refers to the total energy consumed by transmitting all the reports generated in an MWSN and an epoch to the fog node connecting to the MWSN, where the data reports include both the sensed data items and the proof data generated by all the sensor nodes in the MWSN and the epoch.

The parameters used in our simulation and their own default values are shown in Table 2, where the default values of some parameters are set by referencing [19]. In fact, static sensor nodes are also allowed to exist in FMSCSs. In the simulation, we adjust the ratio of the mobile sensor nodes to the total ones in the systems by changing the value of r_{mobile} .

7.2. Simulation Results. This section presents the simulation results of C_{MWSN} and R_{vs} with different settings of r_D , N , and r_{mobile} , respectively. We compare our scheme with VTMSN

[29] and SSSTQ1 [32] in this section. VTMSN, which was proposed in 2015, is the earliest work on securing spatial-temporal Top- k query in FMSCSs, while SSSTQ1 can be considered as the state-of-the-art scheme proposed for securing spatial-temporal Top- k query in FMSCSs. Figure 3 shows the simulation results of C_{MWSN} under different settings of r_D , N , and r_{mobile} , and Figure 4 illustrates the simulation results of R_{vs} with different settings of r_D , N , and r_{mobile} , respectively. From Figure 3, we can see that the C_{MWSN} lines of STQ-SCS are all lower than those of VTMSN and SSSTQ1. This indicates that our proposed scheme STQ-SCS is more energy-efficient than the other two schemes. The C_{MWSN} lines in Figures 3(a) and 3(b) are on an upward trend because the quantity of sensed data items rises as r_D or N becomes larger and larger, which causes the increase of the proof data, while those in Figure 3(c) are on a downward trend as r_{mobile} rises from 0 to 1 because the sensor nodes are assumed to generate sensed data items only when they are

static or arrive at their target locations and the quantity of the sensed data items and the corresponding proof data must decrease when more sensor nodes are set to be mobile.

Thanks to the technology of virtual-location construction proposed in this paper, fewer bits of location information are included in the proof data in STQ-SCS than the other two schemes, which decrease the ratio of the proof data to the whole data including both sensed data items and their proof. From Figure 4, we can see that the values of R_{vs} of STQ-SCS are all under 12% which is within the acceptable range in real applications and also lower than those of the other two schemes.

8. Conclusions

This paper presents a privacy-preservation and integrity-verification scheme named STQ-SCS for fine-grained spatial-temporal Top- k query in FMSCSs. Thorough security analysis shows that STQ-SCS can make the end users in FMSCSs obtain the query results of fine-grained spatial-temporal Top- k queries without disclosing the privacy of both the sensed data items and their scores, considering that the fog nodes and the cloud servers are not trustworthy. Meanwhile, the security analysis also shows that, under the security model described in this paper, the end users in FMSCSs can detect the incomplete Top- k query results with a 100% successful rate based on our scheme STQ-SCS. Simulation results demonstrate that STQ-SCS is much more efficient than the related state-of-the-art schemes, and can be well used in FMSCSs in real applications.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Authors' Contributions

Jie Min conceptualized the study and wrote the original draft of the manuscript and was responsible for methodology; Junbin Liang investigated the study; Xingpo Ma performed simulation; Xingpo Ma and Hongling Chen reviewed and edited the manuscript; Xingpo Ma was involved in project administration and supervision; Hongling Chen performed formal analysis.

Acknowledgments

This work was supported by the Natural Science Foundation of China (Grant no. 61972090), the Natural Science Foundation of Hunan Province (Grant no. 2019JJ40406), the Key Specialized Research and Development Project in Henan Province (Grant no. 202102210161), and Planning Subject for the 13th Five Year Plan of National Education Sciences (Grant no. 2019GXJK272).

References

- [1] X. Li, Z. Ma, J. Zheng, Y. Liu, L. Zhu, and N. Zhou, "An effective edge-assisted data collection approach for critical events in the sdwn-based agricultural Internet of Things," *Electronics*, vol. 9, no. 6, p. 907, 2020.
- [2] A. Liu, X. Liu, and J. Long, "A trust-based adaptive probability marking and storage traceback scheme for wsns," *Sensors*, vol. 16, no. 4, p. 451, 2016.
- [3] Y. Sun, D. Rehfeldt, M. Brazil, D. Thomas, and S. Halgamuge, "A physarum-inspired algorithm for minimum-cost relay node placement in wireless sensor networks," *IEEE/ACM Transactions on Networking*, vol. 28, no. 2, pp. 681–694, 2020.
- [4] S. Misra, S. Chatterjee, and M. S. Obaidat, "On theoretical modeling of sensor cloud: a paradigm shift from wireless sensor network," *IEEE Systems Journal*, vol. 11, no. 2, pp. 1084–1093, 2017.
- [5] T. Aamir, H. Dong, and A. Bouguettaya, "Trust in social-sensor cloud service," in *Proceedings of the 2018 IEEE International Conference on Web Services (ICWS)*, pp. 359–362, Seattle, WA, USA, June 2018.
- [6] Q. Zeng, Q. Duan, M. Shi, X. He, and M. M. Hassan, "Design framework and intelligent in-vehicle information system for sensor-cloud platform and applications," *IEEE Access*, vol. 8, pp. 201675–201685, 2020.
- [7] A. Roy, S. Misra, and F. Nait-Abdesselam, "Range-price trade-off in sensor-cloud for provisioning sensors-as-a-service," *IEEE Transactions on Cloud Computing*, p. 1, 2020.
- [8] X. Wei and L. Wu, "A new proposed sensor cloud architecture based on fog computing for internet of things," in *Proceedings of the 2019 International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*, pp. 615–620, Atlanta, GA, USA, July 2019.
- [9] M. Bazm, M. Lacoste, M. Südholt, and J. Menaud, "Secure distributed computing on untrusted fog infrastructures using trusted linux containers," in *Proceedings of the 2018 IEEE International Conference on Cloud Computing Technology and Science (CloudCom)*, pp. 239–242, Nicosia, Cyprus, December 2018.
- [10] X. Yan, W. W. Y. Ng, B. Zeng et al., "Verifiable, reliable, and privacy-preserving data aggregation in fog-assisted mobile crowdsensing," *IEEE Internet of Things Journal*, p. 1, 2021.
- [11] J. Ye and J. Wang, "Secure outsourcing of modular exponentiation with single untrusted server," in *Proceedings of the 2015 18th International Conference on Network-Based Information Systems*, pp. 643–645, Taipei, Taiwan, September 2015.
- [12] K. N. Sevis and E. Seker, "Survey on data integrity in cloud," in *Proceedings of the 2016 IEEE 3rd International Conference on Cyber Security and Cloud Computing (CSCloud)*, pp. 167–171, Beijing, China, June 2016.
- [13] H. Li, Y. Yang, Y. Dai, S. Yu, and Y. Xiang, "Achieving secure and efficient dynamic searchable symmetric encryption over medical cloud data," *IEEE Transactions on Cloud Computing*, vol. 8, no. 2, pp. 484–494, 2020.
- [14] X. Meng, H. Zhu, and G. Kollios, "Top- k query processing on encrypted databases with strong security guarantees," in *Proceedings of the 2018 IEEE 34th International Conference on Data Engineering (ICDE)*, pp. 353–364, Paris, France, April 2018.
- [15] H. Quan, B. Wang, Y. Zhang, and G. Wu, "Efficient and secure top- k queries with top order-preserving encryption," *IEEE Access*, vol. 6, pp. 31525–31540, 2018.

- [16] S. Su, Y. Teng, X. Cheng, K. Xiao, G. Li, and J. Chen, "Privacy-preserving top- k spatial keyword queries in untrusted cloud environments," *IEEE Transactions on Services Computing*, vol. 11, no. 5, pp. 796–809, 2018.
- [17] D. Negi, S. Ray, and R. Lu, "Pystin: enabling secure lbs in smart cities with privacy-preserving top- k spatial-textual query," *IEEE Internet of Things Journal*, vol. 6, no. 5, pp. 7788–7799, 2019.
- [18] X. Ding, P. Liu, and H. Jin, "Privacy-preserving multi-keyword top- k similarity search over encrypted data," *IEEE Transactions on Dependable and Secure Computing*, vol. 16, no. 2, pp. 344–357, 2019.
- [19] R. Zhang, J. Shi, Y. Liu, and Y. Zhang, "Verifiable fine-grained top- k queries in tiered sensor networks," in *Proceedings of the 2010 IEEE INFOCOM*, pp. 1–9, 2010.
- [20] X. Liao, J. Li, and Y. Lei, "Secure and efficient top- k query processing in two-tiered sensor network," *Journal of Computer Research and Development*, vol. 50, no. 3, pp. 490–497, 2013.
- [21] R. He, H. Dai, G. Yang, T. Wang, and J. Bao, "An efficient top- k query processing with result integrity verification in two-tiered wireless sensor networks," *Mathematical Problems in Engineering*, vol. 2015, Article ID 538482, 8 pages, 2015.
- [22] D. Hua, Y. Geng, X. Fu, and Z. Qiang, "EVTQ: an efficient verifiable top- k query processing in two-tiered wireless sensor networks," in *Proceedings of the 2013 IEEE 9th International Conference on Mobile Ad-hoc and Sensor Networks*, pp. 206–211, Dalian, China, December 2013.
- [23] J. Liang, C. Jiang, X. Ma, G. Wang, and X. Kui, "Secure data aggregation for top- k queries in tiered wireless sensor networks," *Adhoc & Sensor Wireless Networks*, vol. 32, no. 1/2, pp. 51–78, 2016.
- [24] R. Li, A. X. Liu, S. Xiao, H. Xu, B. Bruhadeshwar, and A. L. Wang, "Privacy and integrity preserving top- k query processing for two-tiered sensor networks," *IEEE/ACM Transactions on Networking*, vol. 25, no. 4, pp. 2334–2346, 2017.
- [25] R. Li, Y. Lin, Y. Yi, S. Xiong, and S. Ye, "Security top- k query protocol in two layer sensor networks," *Journal of Computer Research and Development*, vol. 49, no. 9, pp. 1947–1958, 2012.
- [26] C. M. Yu, Y. T. Tsou, C. S. Lu, and S. Y. Kuo, "Practical and secure multidimensional query framework in tiered sensor networks," *IEEE Transactions on Information Forensics and Security*, vol. 6, no. 2, pp. 241–255, 2011.
- [27] W. Chen, L. Yu, and D. Gao, "A privacy preserving histogram aggregation algorithm with integrity verification support," *Chinese Journal of Electronics*, vol. 42, no. 11, pp. 2268–2272, 2014.
- [28] X. Kui, J. Feng, X. Zhou, H. Du, and X. Ma, "Securing top- k query processing in two-tiered sensor networks," *Connection Science*, vol. 33, no. 1, pp. 1–19, 2020.
- [29] F. Liu, X. Ma, J. Liang et al., "Verifiable top- k query processing in tiered mobile sensor networks," *International Journal of Distributed Sensor Networks*, vol. 11, no. 10, Article ID 437678, 2015.
- [30] H. Wu and L. Wang, "Efficient and secure top- k query processing on hybrid sensed data," *Mobile Information Systems*, vol. 201610 pages, 2016.
- [31] X. Ma, X. Liu, J. Liang et al., "A comparative study on two typical schemes for securing spatial-temporal top- k queries in two-tiered mobile wireless sensor networks," *Sensors*, vol. 18, no. 3, p. 871, 2018.
- [32] X. P. Ma, J. B. Liang, J. X. Wang et al., "Secure fine-grained spatio-temporal top- k queries in tmwsns," *Future Generation Computer Systems*, vol. 86, pp. 174–184, 2018.
- [33] J. Li, Z. Guan, X. Du, Z. Zhang, and Z. Zhou, "A low-latency secure data outsourcing scheme for cloud-wsn," in *Proceedings of the 2017 IEEE Wireless Communications and Networking Conference (WCNC)*, pp. 1–6, San Francisco, CA, USA, March 2017.
- [34] B. Nagaraju and P. Ramkumar, "A new method for symmetric key cryptography," *International Journal of Computer Applications*, vol. 142, no. 8, pp. 36–39, 2016.
- [35] E. Khoury, M. Medlej, C. A. Jaoude, and C. Guyeux, "Novel order preserving encryption scheme for wireless sensor networks," in *Proceedings of the 2018 IEEE Middle East and North Africa Communications Conference (MENACOMM)*, pp. 1–6, Jounieh, Lebanon, April 2018.
- [36] S. Jangirala, A. K. Das, M. Wazid, and A. V. Vasilakos, "Designing secure user authentication protocol for big data collection in iot-based intelligent transportation system," *IEEE Internet of Things Journal*, vol. 8, p. 1, 2020.
- [37] S. Verma, R. Choubey, R. Soni, and P. Ogi, "An efficient developed new symmetric key cryptography algorithm for information security," *International Journal of Emerging Technology and Advanced Engineering*, vol. 2, no. 7, pp. 18–21, 2012.
- [38] S. Roy, J. Karjee, U. Rawat, N. Dayama Pratik, and N. Dey, "Symmetric key encryption technique: a cellular automata based approach in wireless sensor networks," *Procedia Computer Science*, vol. 78, pp. 408–414, 2016.
- [39] M. Bala Krishna and M. N. Doja, "Deterministic k -means secure coverage clustering with periodic authentication for wireless sensor networks," *International Journal of Communication Systems*, vol. 30, no. 4, pp. 1–16, 2017.

Research Article

A Detection Approach for Vulnerability Exploiter Based on the Features of the Exploiter

Jinchang Hu ^{1,2}, Jinfu Chen ¹, Sher Ali ¹, Bo Liu ¹, Jingyi Chen ¹, Chi Zhang ¹,
and Jian Yang ¹

¹School of Computer Science and Communication Engineering, Jiangsu University, Zhenjiang, China

²Command and Control Engineering College, Army Engineering University of PLA, Nanjing, China

Correspondence should be addressed to Jinfu Chen; jinfuchen@ujs.edu.cn

Received 24 February 2021; Revised 17 April 2021; Accepted 6 May 2021; Published 22 May 2021

Academic Editor: Ke Gu

Copyright © 2021 Jinchang Hu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

With the wide application of software system, software vulnerability has become a major risk in computer security. The on-time detection and proper repair for possible software vulnerabilities are of great importance in maintaining system security and decreasing system crashes. The Control Flow Integrity (CFI) can be used to detect the exploit by some researchers. In this paper, we propose an improved Control Flow Graph with Jump (JCFG) based on CFI and develop a novel Vulnerability Exploit Detection Method based on JCFG (JCFG-VEDM). The detection method of the exploit program is realized based on the analysis results of the exploit program. Then the JCFG is addressed through combining the features of the exploit program and the jump instruction. Finally, we implement JCFG-VEDM and conduct the experiments to verify the effectiveness of the proposed method. The experimental results show that the proposed detection method (JCFG-VEDM) is feasible and effective.

1. Introduction

With the development of society, the computer network has taken roots in every direction of the society as an essential part of modern life. However, there is no effective detection method for existing malicious programs in the network. People enjoy the convenience brought by computer technology but do not have an effective method to prevent the exploit programs [1–3]. At present, researchers have made some achievements in this field, but in the face of the endless stream hijacking attacks, the current exploit detection methods are still lacking pertinence [4–6]. Therefore, in view of the current vulnerability exploit attacks and their variants, we propose an improved Control Flow Graph (CFG) incorporating the features of the exploit programs. It is of great significance to the detection of exploit programs, beneficial to researches on the detection method of exploit programs. We also propose a Vulnerability Exploit Detection Method based on CFG with Jump (JCFG) (JCFG-VEDM).

The main contributions of this paper are as follows:

- (1) The feature definition of the exploit for the abnormal jump is proposed based on empirical analysis.
- (2) Under the premise of understanding the attack principles of existing exploits, we analyze the features of exploits, integrate the feature information into the CFG, and add pointer-related concepts. In addition, we propose the JCFG by combining the Control Flow Integrity (CFI) detection method.
- (3) Based on the research of vulnerability features, this paper also proposes the JCFG-VEDM and focuses on the JCFG based on the features of the vulnerability exploit generation and detection algorithm of exploit programs. The experiments have shown that JCFG-VEDM has good feasibility and effectiveness in detecting abnormal jumps of exploit programs.

The rest of the paper is organized as follows. In Section 2, we describe the related work. In Section 3, we introduce the analysis method of the exploit based on program features and Section 4 proposes the vulnerability exploit detection method based on the features of vulnerability exploit. The

experimental analysis is reported in Section 5. Conclusions are presented in Section 6.

2. Related Work

The current exploit detection methods mainly use the control flow of a program to detect and protect the program, including CFI and taint analysis.

The original intention of CFI is to eliminate control flow hijacking attacks. In 2005, CCS (ACM Conference on Computer and Communications Security) published a paper called “Control Flow Integrity (CFI)” proposing the concept of CFI [7, 8].

CFI detection is divided into fine-grained CFI and coarse-grained CFI. The fine-grained CFI is proposed primarily, obtaining the corresponding CFG through static analysis of the program, calculating the destination address that the jump instruction may reach, and assigning an address ID to each address. Whenever the program jumps, it is logically checked to check whether the jump is a legal address [9, 10]. For example, XFI [11] modularized the program by combining memory access mechanisms to protect and monitor the program while it is running. However, due to the fine-grained CFI imposes too much overhead on the system, it is difficult to implement. The CFI program proposes a simplified version of the solution, which is called coarse-grained CFI [12, 13]. Coarse-grained CFI neither needs to obtain the CFG of the program nor needs to assign a corresponding address ID to each address where the program jumps. It only needs to perform a static analysis on the program and calculate the legal transfer addresses through the corresponding rules [14–16]. For example, CCFIR (Compact Control Flow Integrity and Randomization) [17] collects all legal transfer addresses together, and the program can only jump between these legal addresses. CFIMon [18] uses static analysis of the program to obtain the legal access addresses and then uses the score tracking storage mechanism (LBR, etc.) in the processor to detect the program and analyze the CFI of the program in real time. For all that, these coarse-grained CFI detection methods can reduce system overhead, the jump instruction and the return instruction cannot be one-to-one correspondence due to the fact that each jump address is not assigned a corresponding address ID, which leads to a call instruction that can enter any function. In the beginning, a vulnerability is created.

At present, among the various techniques for program analysis, researchers in the field of program analysis prefer stain propagation analysis, which is combined with the analysis technology to analyze the program for a more accurate program analysis report. Stain analysis is divided into static analysis and dynamic analysis [19]. Static Taint Analysis is to analyze the program statically without running the program to detect whether the data can be transmitted from the source of the taint to the spot [20–22], whereas Dynamic Taint Analysis is to detect whether the data can be transmitted from the source to the aggregation point while the program is running [23, 24].

In recent years, the academic circles mainly use CFI, stain analysis, and other detection schemes to detect the

exploit [25, 26]. These schemes detect the abnormal control flow during running the program and have achieved certain results in practical application. However, the integrity detection of control flow needs to deal with the program at code level. Taint analysis mainly monitors the dynamic execution process of the program and sends out an alarm when the tainted data are used abnormally. The detection results have a certain degree of error because of the method’s own limitations. These methods need instrumentation in the program and have excessive system overhead. Therefore, in view of the abnormal jump in the exploit, it is of great significance to deeply analyze the features of the exploit and provide the definition and unified formal description of the features of the exploit, which can further promote the research on the detection of the exploit, and the approach that we proposed does not need instrumentation, making security researchers more convenient to detect the exploits.

3. Analysis Method of Exploit Based on Program Features

The program features of the exploit rely mainly on the result of abstracting the program features of the exploit identified by the program feature. So far, the research on the program features of the exploit is not mature enough. The feature definition and formal description of the exploit are beneficial to the research of the exploit detection method.

This section first analyzes the features of the exploit and then the JCFG is proposed. Finally, the exploit is formalized through JCFG in this section.

3.1. Definition of Exploit Features

3.1.1. Lexical, Grammatical, and Semantic Features. There are diversified ways of exploiting exploits, with the main modality referred to the use of some specific instructions to achieve the attack from the perspective of assembly code. Therefore, we analyze the exploit from three aspects: lexicon, grammar, and semantics. For example, for each transfer instruction, when the transferred address does not exist in the legal transfer address, the node of the transfer instruction constitutes a dangerous node of exploit. An exploit example with C source code is shown in Figure 1.

This program compares the string in the file with PASSWORD, verifies whether it is consistent, and outputs the result. In this program, there is a vulnerability in the buff array in the verification function. By overwriting its return address, the program can jump to the starting address of the shellcode to perform related operations. Its corresponding assembly code flowchart is shown in Figure 2.

In Figure 2, the exploit can direct the control flow of the program to the address of the shellcode by overwriting the return address of the verify function. This is the dangerous node of exploit. Figure 2 mainly shows that the structure block of the verify function is mainly shown without some other system function calls. The call, jmp, jz, jnz, ret, and so on are instructions in the assembly code which constitute the grammatical features of the exploit. Through summarizing

```

1. #include <stdio.h>
2. #include <windows.h>
3. int verity (char *password)
4. {
5.     int result;
6.     char buff [8];
7.     result = strcmp (password, password);
8.     strcpy (buff, password);
9.     return result;
10. }
11. void main ()
12. {
13.     int flag = 0;
14.     char password [1024];
15.     FILE * fp;
16.     if (! (fp = fopen ("password.txt", "rw+")))
17.     {
18.         exit (0);
19.     }
20.     fscanf (fp, "%s", password);
21.     flag = verity (password);
22.     fclose (fp);
23.     if (flag)
24.     {
25.         printf ("false.");
26.     }
27.     else
28.     {
29.         printf ("true.");
30.     }
31. }

```

FIGURE 1: An exploit example with C source code.

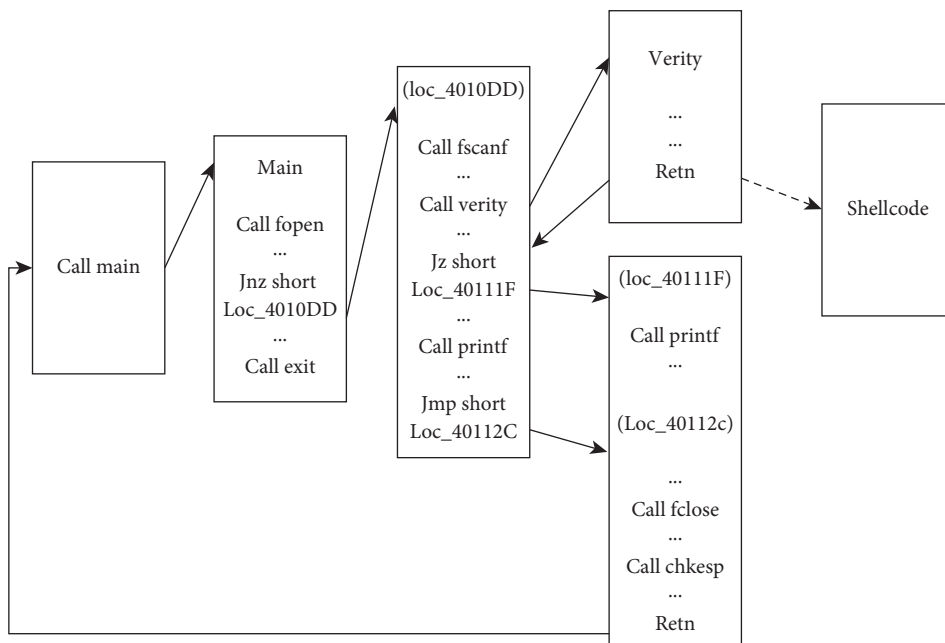


FIGURE 2: The flowchart of the vulnerable assembly code.

the grammatical features of these instructions, the following points are obtained: (1) function call instruction Call; (2) address transfer instruction JXX; (3) return instruction Return. In this paper, we refer to the grammatical features of these exploits as the dangerous element of exploit represented by σ . The dangerous node of exploit must contain the dangerous element of exploit. Therefore, we mainly analyze the address of the dangerous element of an exploit in the program to determine whether it is dangerous.

The previously mentioned dangerous element of exploit is shown in Figure 2. By analyzing the location of the dangerous element of exploit, it can be judged whether it belongs to the dangerous node of exploit, when the block program pointed by the dotted line of the program is executed in Figure 2. In this paper, a collective symbol D is established to contain all the dangerous nodes of exploit in the program. Afterwards, we will further analyze the dangerous node of exploit to determine whether they belong to the exploit nodes. In addition, the set of exploit nodes is denoted as V .

3.1.2. Definition Description. Albeit many ways to exploit, generalizing them from the perspectives of grammar, morphology, and semantics can always find a certain similarity. In this paper, the program features of the exploit are denoted as μ , and the constraint of the exploit is denoted as C . The program features of the exploit in this paper are described in Definition 1.

Definition 1. Program features of the exploit μ : $\mu(\text{Vul}) = \{D, C\}$. Vul represents the type of vulnerability exploited by the exploit. Prog represents the program containing instructions for the dangerous element of exploit σ , that is, the dangerous node of exploit. D is the collection of these dangerous nodes of exploit, such that, $D = \{d_1, d_2, \dots, d_n\}$. C represents the relevant vulnerability exploit constraints that the features of the exploit program need to meet, such that $C = c_1 || c_2 || \dots || c_j || \dots || c_n$. For an exploit, the basic constraint Bc_j and the additional constraint Tc_j of the exploit need to be satisfied, meaning $c_j = Bc_j \wedge Tc_j$.

In the program feature μ of the exploit, D describes the performance and syntax features of the exploit, and C describes the semantic feature of the exploit. And, for the program feature of the exploit, it also has the following properties.

Property 1. The number of nodes is limited for a program. Accordingly, the number of dangerous nodes of exploit is also limited.

Property 2. For the dangerous node of exploit, the node must contain the dangerous element of exploit.

3.2. Formalization of Exploit Features. The feature form of the exploit refers to the formal expression of the feature of the exploit, which provides a strong foundation for describing the exploit in more detail and facilitates the research

on the detection of the exploit. The main research object of this section is the exploit of abnormal jump.

3.2.1. Control Flow Graph Based on Jump (JCFG). At present, most of the detection methods for exploit used by researchers are to design corresponding exploit detection algorithms through the CFG of the program for detecting the exploit. The main detection method for exploit is CFI, which is divided into fine-grained CFI and coarse-grained CFI. For fine-grained CFI, it allocates a unique ID for each instruction jump and adds the detection function to the program. The system overhead is exceedingly large, and the efficiency cannot meet real needs. Therefore, the researchers proposed the coarse-grained CFI which does not need to assign a unique ID to each jump instruction but needs to detect whether the address of each jump is in the legal address set. However, the coarse-grained CFI has a certain impact on the accuracy of detection. Therefore, based on the strengths and weaknesses of the above two exploit detection methods, this paper proposes a new control flow graph based on the CFG by combining the features of the exploit, called the Control Flow Graph-based Jump (JCFG). In JCFG, only dangerous nodes of exploit are included. For each dangerous node of exploit, the node in JCFG mainly contains the following attributes: (1) instruction type, including jmp, call, jz, jnz, and ret; (2) the name of the called function; (3) jump address. These attributes are recorded as the feature attributes of the node.

Definition 2. Control Flow Graph based on Jump (JCFG): $JCFG = (D, E, R, \text{Begin}, \text{End})$. D represents the set of dangerous nodes of exploit contained in JCFG. For the dangerous node of exploit d in the set, $d = (\text{id}, \text{attr}, \text{next}_{\text{id}})$, id represents the number of the node in JCFG, attr represents the feature attribute of the node, and next_id represents the node that the current node points to. For next_id, there may be a forked path, so the first node pointed to is marked as *first, and the second node pointed to is marked as *second. For node attributes, address represents the current address of the instruction, attrName represents the name of the instruction, funcName represents the name of the function, and jAddress represents the jump destination address. E represents the combination of edges, used to express the direction relationship between nodes. R represents the set of return addresses.

$$\text{attr} = (\text{address}, \text{attrName}, \text{funcName}, \text{jAddress}). \quad (1)$$

For each call instruction, the address after instruction is called is added to R . Begin is the entry node, and End is the end node of JCFG.

3.2.2. Related Definitions

Definition 3. Call instruction: Call represents call instruction.

Definition 4. Jump instruction: JXX represents the jump instruction, which includes the conditional jump instruction

JCC (where CC represents the character sequence of the test condition type, including jz and jnz) and the unconditional jump instruction jmp.

Definition 5. Return instruction: Return represents return instruction, including RETN and RETF return instructions. RETN represents return from the subroutine transferred in the segment, and RETF represents return from the subroutine transferred in between segments.

Definition 6. Return address set: each time a function call instruction is executed, the address following the call instruction is stored in the return address set R .

Definition 7. Node judgement: for the program, there may be remerging the two execution paths after jnz, so the judgement is made to avoid duplication. The judgement function is recorded as isSame (jAddress).

Definition 8. JCFG node pointer: $*p$ represents the current JCFG node pointer pointing to the Begin node by default. After the main function is executed, $*p$ points to the first dangerous node of exploit under the Begin node.

3.2.3. Example Analysis. Figure 3 shows the code segment of a simple program. This program is simplified based on the code segment of the exploit given in Figure 1. Its specific assembly code flow graph is shown in Figure 4, and the JCFG formed is shown in Figure 5.

First, the Begin node is obtained according to the main function, and the next address of the call instruction in the return address set R is stored. Then the conditional jump instruction in the JXX instruction is matched, forming the d_1 node through it. Each time the jump instruction JXX is matched, the same node determination function is called, and the same node is checked first. If it exists, the node pointing to it is directly pointed to the existing node. If it does not exist, then it is examined whether it exists if the node with the same jump destination address exists, and the next node id of the jump destination address path is added to the next_id of the node. The d_1 node has two successor nodes. The process of d_1 node is as follows: (1) select a successor path of d_1 ; (2) read the next call instruction; (3) put its next address into the return address set R ; (4) get the d_2 node, read two call instructions and in turn, get d_3 and d_4 nodes, and put the next address of their call instruction into the return address set. The next matched instruction is the conditional jump instruction in the JXX instruction, through which the d_5 node is formed. The d_5 node has two successor nodes. The process of d_5 node is similar to d_1 node, generating the d_6 node. The subsequent matched instruction is the unconditional jump instruction in the two JXX instructions. The d_7 node and the d_8 node are formed, respectively, and then the matched instruction is found to exist through the same node judgement function, so the d_8 node points to the d_1 node. Then, it returns to another instruction path of the previous d_5 node. Followed by a call instruction and a JXX instruction, the corresponding d_9 and d_{10} nodes

```

1. #include <stdio.h>
2. #include <windows.h>
3. int verity (char* password)
4. {
5.     int result = strcmp (password, PASSWORD);
6.     return result;
7. }
8. void main ()
9. {
10.    int flag = 0;
11.    char password [1024];
12.    while (1)
13.    {
14.        printf ("input password: ");
15.        scanf ("%s", password);
16.        flag = verity (password);
17.        if (flag)
18.        {
19.            printf ("false.");
20.        }
21.        else
22.        {
23.            printf ("true.");
24.            break;
25.        }
26.    }
27. }

```

FIGURE 3: Simple program code segment.

are obtained, the following instruction is a Call instruction to get the corresponding d_{11} node, followed by a return instruction Return, and there is only one return address in the return address set R . Therefore, the node is determined as the End node. Backing up again, another instruction path can access the d_1 node, through the same node determination function, d_{11} can be added to the next_id set of d_1 . This is the end and the JCFG graph is generated.

Here is just a brief description of the generation of JCFG. The specific JCFG generation algorithm will be introduced in detail in the next section.

3.3. JCFG Generation Method Based on the Characteristics of the Exploit

3.3.1. Control Flow Graph Generation Method by IDA. For the exploit to be detected, with the static analysis of the exploit, the exploit is imported into IDA, getting its assembly code and reading its instructions. The efficiency of generating JCFG directly by extracting instructions from the assembly code of the program to be tested is extremely low. Therefore, the process of generating the JCFG is as follows: (1) use IDA scripts to generate corresponding CFG; (2) use the CFG that generated by IDA to filter out some unimportant instructions and retain the required Call, JXX, Return, and other key Command; (3) process the generated CFG to obtain the required JCFG. The node information structure of the CFG is shown in Figure 6.

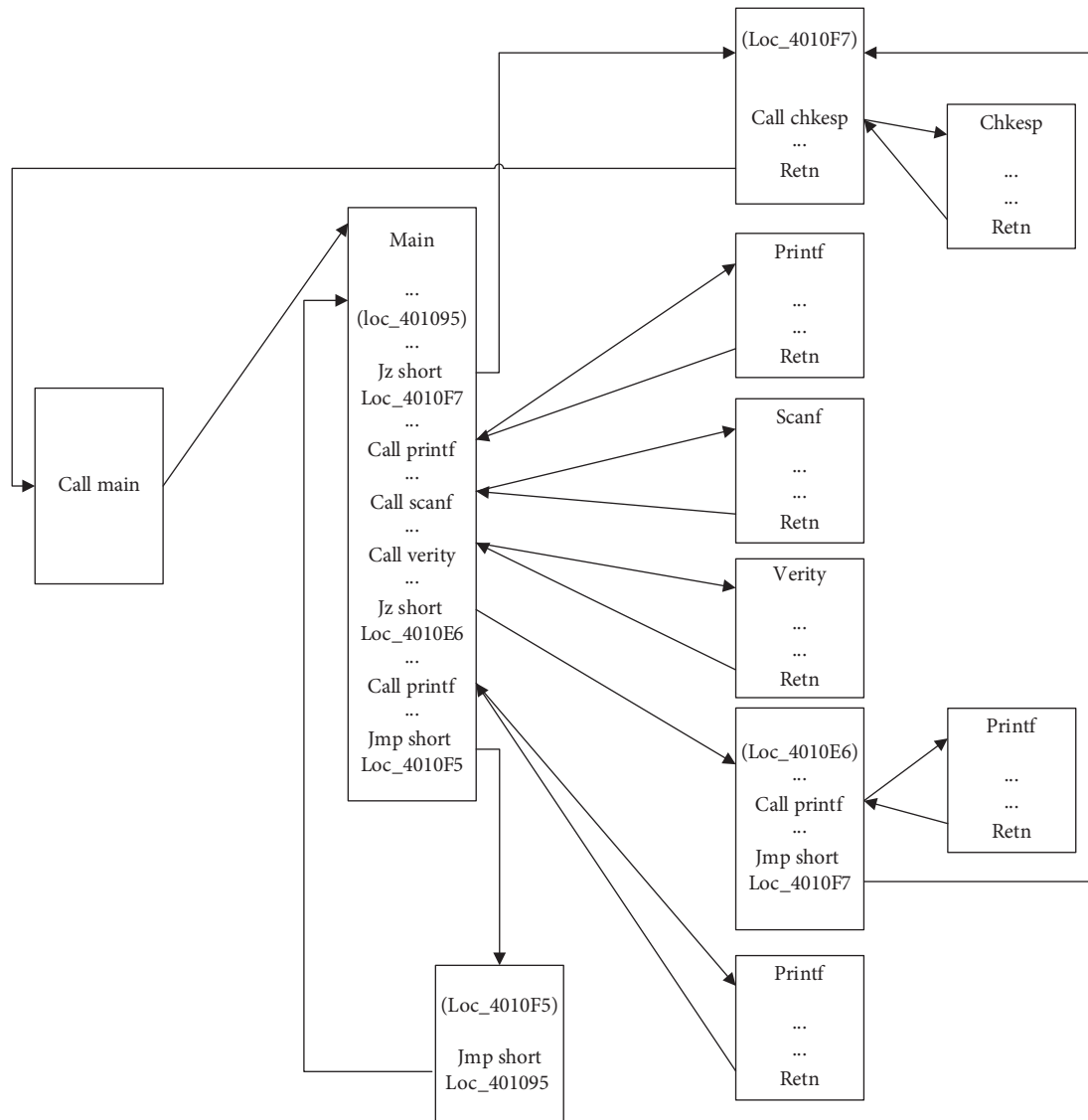


FIGURE 4: The flowchart for assembly code.

The CFG generation algorithm is shown below. The algorithm uses a recursive method to generate the CFG. The input is the assembly code of the exploit, and the output is the SQL file containing the CFG structure and all CFG node information. Algorithm 1 shows how to further process the statements containing key instructions in the assembly code of the exploit. Lines 7–12 are for processing the statements containing Call instructions, and lines 13–15 are for processing the statements containing Return. Lines 15–31 are to process the instruction statement containing JXX. The time-consuming is mainly on the program traversal process of the algorithm, with the time complexity $O(n)$, where n is the number of assembly code lines of the exploit.

3.3.2. Node Information Extraction Method. The node information extraction method mainly extracts the node information contained in the generated CFG. The instruction information in the exploit has been filtered out of a large part of

the noncritical information in the process of generating the CFG. Here, the corresponding processing is mainly for the filtered information, which is convenient for use when generating the JCFG. In Figure 6, we can see the data structure design of the node attributes of the CFG. In Figure 7, we have further extracted the instruction information and subdivided it for the data structure design of the node attributes of the CFG, which are the instruction name, function name, and destination address. There are two types of command names, namely, the Call command and the JXX command mentioned.

3.3.3. JCFG Generation Method. The JCFG generated in the algorithm is recursive. The input is the CFG generated by the IDA script, and the output is the JCFG. Algorithm 2 further processes the node information of each node in the CFG. Lines 5–11 are for processing the node information of CFG nodes containing Call instructions, and lines 12–28 are for CFG nodes containing JXX instructions to process the node

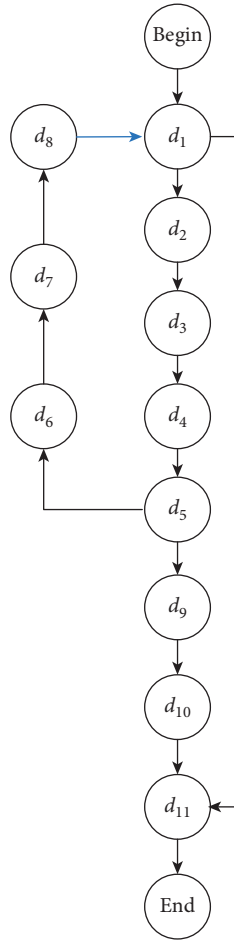


FIGURE 5: JCFG of a simple code segment.

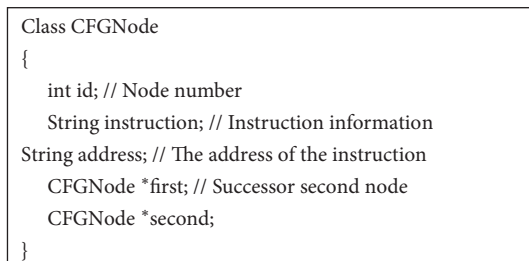


FIGURE 6: The data structure for CFG node attribute.

information. Among them, lines 13–19 are for the node information of the CFG node of the direct jump instruction in the JXX instruction, and lines 20–27 are for the indirect jump instruction in the JXX instruction. The operation time of this algorithm is mainly spent on traversing the CFG graph nodes. As the number of CFG graph nodes is limited, the time complexity of this algorithm is $O(n)$, where n is the number of nodes in the CFG.

4. Exploit Detection Method Based on JCFG

The focus of this paper is to study the detection methods for the exploit from all aspects of the exploit. In the previous section, JCFG is proposed; in this section, we will use the

JCFG to detect the exploit. This section explains the process of detecting the exploit with JCFG and the further analysis of the nodes in it with the aim of determining whether it meets the constraints of the exploit.

4.1. Exploit Detection Framework. This paper proposes a Vulnerability Exploit Detection Method based on JCFG, denoted as JCFG-VEDM, which is used to detect vulnerabilities. In addition, the proposed JCFG-VEDM is evaluated according to the detection results of this method.

Figure 8 shows the components of the JCFG-VEDM, including the following modules: JCFG generation module, execution module of exploit, and exploit judgement module.

4.2. Related Definitions and Example Analysis

4.2.1. Related Definitions

Definition 9. Function Name Judgement, $CNameJudge(JCFGNode, *q)$: when the program is dynamically executed, the current execution instruction is the Call. Comparing the function name of the node pointed to by the name of the calling function after the call and the node pointer of the current JCFG, if both are consistent, false is returned; otherwise, true is returned.

Definition 10. Jump Address Judgement, $JAddressJudge(JCFGNode, *q)$: when the program is dynamically executed, the current execution instruction is JXX, and the subsequent jump address is compared with the destination address $JAddress$ of the node pointed to by the current JCFG node pointer. The same returns false, and the different returns true.

Definition 11. Return Address Judgement, $RetnJudge(R, *q)$: when the program is dynamically executed, the current execution instruction is the Return, and the address after execution is compared with the uppermost address in the return address set R . The same returns false, and the different returns true.

Definition 12. Instruction containment: $Include(d, instruction)$ to indicate that the currently executed instruction $node$ d contains an instruction that has been defined like the Call, the JXX, or the Return. For example, $(\exists d \in Prog) \wedge (Include(d, Call))$. This means that the current execution instruction node of the Prog has Call.

Definition 13. Program execution pointer: $*q$ represents the instruction node that the program is currently executing.

4.2.2. Abnormal Jump. The execution flow is hijacked during running the program, so that the program executes code that should not be executed, which is called an abnormal jump. According to the feature definition of the exploit, $\mu(AJ) = \{D, C\}$, with a formal description of the vulnerable node D and the program feature constraint condition C of the exploit followed:

```

Input: Exploit/ * Instruction of the exploit */
Output: CFG/ * CFG nodes information stored in the database */
(1)   CFG = new CFG (); /* Initialize CFG */
(2)   Instruction instruction; /* The command information of the current read line */
(3)   Stack jN = new Stack <>(); /* Create a stack to store the number of instruction lines for conditional jumps and path forks
*/
(4)   Stack R = new Stack <>(); /* Create a stack to store the address that should be returned when calling the function */
(5)   int id = 1; /* Record the number of CFG nodes */
(6)   for (int i = 0; i < n; i++) do
(7)     if (instruction.exist (Call)) then
(8)       CFGNode = new CFG (instruction);
(9)       if (!isSame (CFGNode)) then
(10)        CFGAdd (CFGNode);
(11)      id++;
(12)    end if;
(13)  else if (instruction.exist (Return)) then
(14)    Return (R); /* Return the address stored in R */
(15)  else if (instruction.exist (JXX)) then
(16)    if (instruction.exist (jmp)) then
(17)      CFGNode = new CFG (instruction);
(18)      if (!isSame (CFGNode)) then
(19)        CFGAdd (CFGNode);
(20)    else
(21)      Return (jN) /* Return the address stored in jN */
(22)    end if;
(23)  else if (instruction.exist (jnz) or instruction.exist (jz)) then
(24)    CFGNode = new CFG (instruction);
(25)    if (!isSame (CFGNode) or (isSame (CFGNode).second == null)) then
(26)      CFGAdd (CFGNode);
(27)    else
(28)      Return (jN) /* Return the address stored in jN */
(29)    end if;
(30)  end if;
(31) end if;
(32) end for.

```

ALGORITHM 1: CFG generation algorithm.

```

Class JCFGNode
{
  int id; // Node number
  NodeAttr * attr; // Node attributes
  JCFGNode *first;
  JCFGNode *second;
}
Class NodeAttr
{
  String jAddress; // Destination address
  String address; // Address of the instruction
  String attrName; // Command name
  String funcName; // Function name
}

```

FIGURE 7: The data structure for the JCFG node attribute.

(1) Dangerous node of exploit, $D = \{D^{CNameJudge}, D^{JAddressJudge}, D^{RetnJudge}\}$. Among them, $D^{CNameJudge} = \{d | (\exists d \in D) \wedge (\text{Include}(d, \text{Call}))\}$, $D^{JAddressJudge} =$

$\{d | (\exists d \in D) \wedge (\text{Include}(d, \text{JXX}))\}$, $D^{RetnJudge} = \{d | (\exists d \in D) \wedge (\text{Include}(d, \text{Return}))\}$.

Description: $D^{CNameJudge}$ represents the collection of nodes which can call the function name judgement and returns true in the program. $D^{JAddressJudge}$ represents the collection of nodes which can jump address judgement and returns true in the program. And $D^{RetnJudge}$ represents the program return address judgement and returns a collection of nodes that refer to true.

(2) Relevant constraints on exploit features, denoted as C .

$$\begin{aligned}
C_1 = & ((\exists^* p \in D, ^* q \in \text{Prog}) \\
& \wedge \text{Include} (^* p, \text{Call}) \\
& \wedge \text{Include} (^* q, \text{Call}) \\
& \wedge CNameJudge (^* p, ^* q)).
\end{aligned} \tag{2}$$

Description: the constraint C_1 related to the exploit feature indicates the existence of the exploit JCFG

```

Input: CFG
Output: JCFG
(1)   JCFG = new JCFG (); /* Initialize JCFG */
(2)   Stack cN = new Stack <> (); /* Create a stack to store conditional jumps and path fork nodes */
(3)   Stack jN = new Stack <> (); /* Create a stack to store conditional jumps and path fork nodes */
(4)   for (int i = 0; i < n; i++) do
(5)       if (node.instruction.exist (Call)) then
(6)           JCFGNode = new JCFG (nodeAttrExtract (node.instruction, node.adress));
(7)           if (JCFG.exist (JCFGNode)) then
(8)               Return (cN, jN); /* Return the CFG node of the last forked path, and make the current JCFG node become the
CFG node of the last forked path */
(9)       else
(10)          JCFGAdd (JCFGNode);
(11)       end if;
(12)       else if (instruction.exist (JXX)) then
(13)           if (node.instruction.exist (jmp)) then
(14)               JCFGNode = new JCFG (nodeAttrExtract (node.instruction, node.adress));
(15)               if (JCFG.exist (JCFGNode)) then
(16)                   Return (cN, jN);
(17)               else
(18)                   JCFGAdd (JCFGNode);
(19)               end if;
(20)           else if (node.instruction.exist (jnz) or node.instruction.exist (jz)) then
(21)               JCFGNode = new JCFG (nodeAttrExtract (node.instruction, node.adress));
(22)               if (!JCFG.exist (JCFGNode) or JCFG.second == null) then
(23)                   JCFGAdd (JCFGNode);
(24)               else
(25)                   Return (cN, jN);
(26)               end if;
(27)           end if;
(28)       end if;
(29)   end for.

```

ALGORITHM 2: JCFG generation algorithm.

node $*p$ in the dangerous nodes of exploit set D and the existence of the instruction node $*q$ in the detected exploit, and the instruction of the node where $*p$ is located is Call. The instruction of the instruction node where $*q$ is located is also the Call, calling the function name judgement to determine whether there is an abnormal jump.

$$\begin{aligned}
C_2 = & ((\exists *p \in D, *q \in \text{Prog}) \\
& \wedge \text{Include} (*p, \text{JXX}) \\
& \wedge \text{Include} (*q, \text{JXX}) \\
& \wedge \text{JAddressJudge} (*p, *q)).
\end{aligned} \tag{3}$$

Description: the constraint C_2 related to the features of the exploit means the existence of the exploit JCFG node $*p$ in the dangerous nodes of exploit set D , the existence of the instruction node $*q$ in the detected exploit, and the instruction of the node where $*p$ is located which is JXX. The instruction of the instruction node where $*q$ is located is also JXX. Currently, it calls the jump address judgement to determine whether there is an abnormal jump.

$$\begin{aligned}
C_3 = & ((\exists *p \in D, *q \in \text{Prog}) \\
& \wedge \text{Include} (*p, \text{Return}) \\
& \wedge \text{Include} (*q, \text{Return}) \\
& \wedge \text{RetnJudge} (*p, *q)).
\end{aligned} \tag{4}$$

Description: the constraint C_3 related to the features of the exploit means that there is an exploit JCFG node $*p$ in the dangerous nodes of exploit set D and an instruction node $*q$ in the detected exploit. The instruction of the node where $*p$ is located is Return instruction, and the instruction of the instruction node where $*q$ is located is also the Return. Currently, the return address judgement is called to determine whether there is an abnormal jump.

4.2.3. Example Analysis. Figure 9 shows a code segment of an exploit. The exploiting in this program is to overwrite the return address of the strcpy function in the verify function to import the execution flow of the program into the shellcode. In the main function, the program reads the string in the password.txt and compares it with PASSWORD. The JCFG

Input: JCFG, The exploit node to be detected/ * The SQL containing JCFG node information and the node extracted from key instructions of the program during execution */

Output: Result

```

(1)  JCFGNode *p=JCFG.head (); /* Make the JCFG pointer point to the head node of JCFG */
(2)  Node *q=new Node (); /* The instruction node in the execution process, the main function is first located when the
program is executed */
(3)  Stack R=new Stack <> (); /* Create a stack to store the return address */
(4)  vector <stack<VulNode>> Vul; /* Create a stack to record the exploit nodes that generate abnormal jumps */
(5)  begin
(6)    if (q.attrName==p.first.attr.attrName) then
(7)      if (q.attrName==Call) then
(8)        if (CNameJudge (p.first, q)) then/*
(9)          Vul.add (temp);
(10)         enf if;
(11)       else if (q.attrName==JXX) then
(12)         if (JAdressJudge (p.first, q)) then
(13)           Vul.add (temp);
(14)         end if;
(15)       else
(16)         if (RetnJudge (R, * q)) then
(17)           Vul.add (temp);
(18)         end if;
(19)     else if (q.attrName==p.second.attr.attrName) then
(20)     else
(21)       Vul.add (temp);
(22)     end if;
(23) end

```

ALGORITHM 3: Vulnerability Exploit Detection Method based on JCFG (JCFG-VEDM).

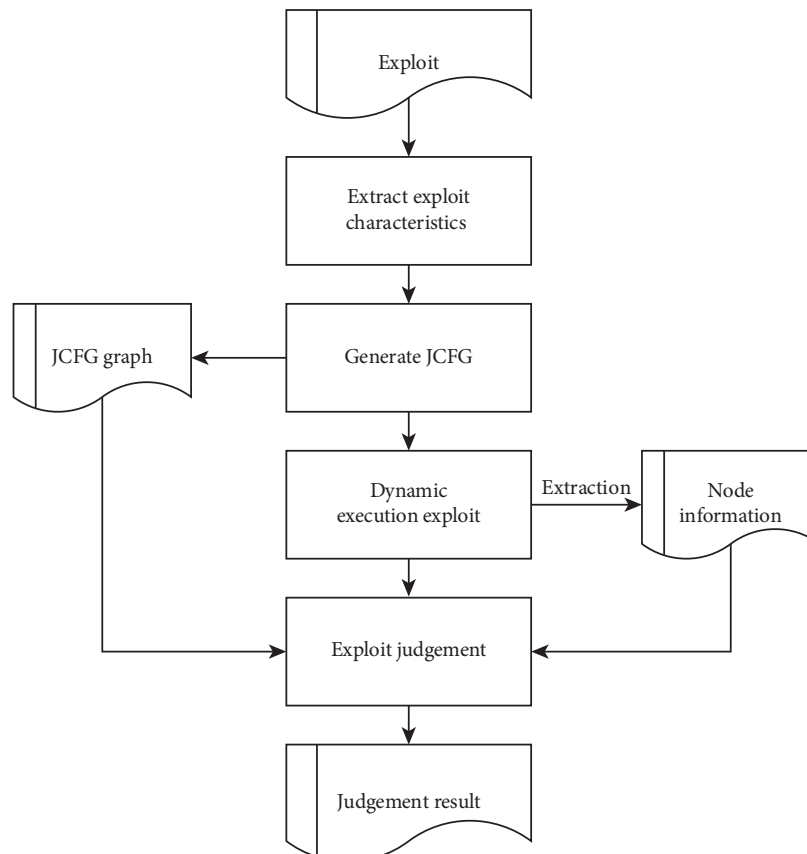


FIGURE 8: Exploit Detection Framework based on JCFG.

obtained by static analysis of the program is shown in Figure 10.

In Figure 10, the part of the JCFG that will cause the abnormal jump in the exploit is mainly for analysis. After the program has executed a series of instructions in the d_j node, it should return from the entered `_strcpy` function. The last node executed before returning is the d_j node, while the node executed afterwards should be the d_{j+1} node. The node information corresponding to these nodes is shown in Figure 11. However, when the exploit is officially executed, the return address of this program is overwritten because the program reads the information in the txt, and the execution flow is imported into the shellcode. When the program generates an abnormal jump, after the program executes the d_j node, the next instruction read is `Call MessageBoxA`, and the information of the $*q$ execution node is shown in Figure 11. The $*p$ node in the current JCFG is at the position of the d_j node. By comparing the node information of the subsequent nodes, the $*q$ node and the $*p$ node in the JCFG, it will be found that both do not match, with an abnormal jump generated.

4.3. Vulnerability Exploit Detection Method Based on JCFG.

For the exploit to be detected, it is dynamically analyzed after static analysis is finished. Ollydbg is an extremely popular program dynamic analysis tool, through which the program can be dynamically analyzed very conveniently. When the program is dynamically analyzed, corresponding instruction nodes are generated for the key instructions in the execution process. The key instructions include the previously defined `Call`, `JXX`, and `Return`. The node attribute data structure design of the instruction node is shown in Figure 12. For key instructions in the execution process, the corresponding instruction nodes are generated and compared with the execution nodes in the JCFG to determine whether abnormal jumps occurred.

The specific description of the Vulnerability Exploit Detection Method based on JCFG is shown in Algorithm 3, and the part about extracting key instruction information during execution is omitted from the algorithm. The algorithm mainly shows the detection function of the exploit. According to the input execution instruction node and the node pointer of the JCFG, the detection result is obtained by matching. In lines 6–18, these are to match the subsequent first node of the execution instruction node $*q$ and the node pointer $*p$ of the JCFG. Lines 7–10 are the check function for the key instruction `Call`. Lines 11–14 are for the key instruction `JXX` check, and lines 15–18 are check functions for the key instruction `Return`. Then, it matches the subsequent second node of the execution instruction node $*q$ and the node pointer $*p$ of the JCFG. The specific operation is like the previous operation. Lines 20–22 indicate that the execution instruction node $*q$ does not match the subsequent first node and second node of the node pointer $*p$ of the JCFG, so the current execution instruction node and the node of the JCFG are stored in the exploit node stack. Most of the execution time of the algorithm is spent on the step-by-step reading of the execution instructions. The exploit has

```

1. #include <stdio.h>
2. #include <windows.h>
3. int verity (char * password)
4. {
5.     char buff [8];
6.     int result = strcmp (password, PASSWORD);
7.     strcpy (buff, password);
8.     return result;
9. }
10. void main ()
11. {
12.     int flag = 0;
13.     char password [1024];
14.     FILE * fp;
15.     LoadLibrary ("user32.dll");
16.     if (! (fp = fopen ("password.txt", "rw+")))
17.     {
18.         exit (0);
19.     }
20.     fscanf (fp, "%s", password);
21.     flag = verity (password);
22.     if (flag)
23.     {
24.         printf ("false.");
25.     }
26.     else
27.     {
28.         printf ("true.");
29.     }
30.     fclose (fp);
31. }

```

FIGURE 9: Exploit code segment.

limited instruction statements, so the execution instruction nodes formed are less than the number of instruction statements of the exploit. The time complexity of this algorithm is $O(n)$, where n is the total number of nodes that generate and execute instructions.

5. Experimental Analysis

This chapter mainly elaborates the various information of the experiment, which includes the various indicators needed for the experiment, the prepared experimental plan, and the experimental results.

5.1. Experimental Program. This section selects some typical exploits for detection which cover a variety of attack types, including `ret-to-libc`, `ROP`, and `JIT Spraying`, and will give the comparison of JCFG with `DEP` protection strategy of the system and `ASLR` address randomization protection strategy. The exploits to be detected are shown in Table 1.

First, the IDA script is used to perform static analysis on the exploit, extracting the obtained assembly code, the key instruction information, and the corresponding CFG generated, which is stored in the database. Then by further extracting the node information in the CFG, the

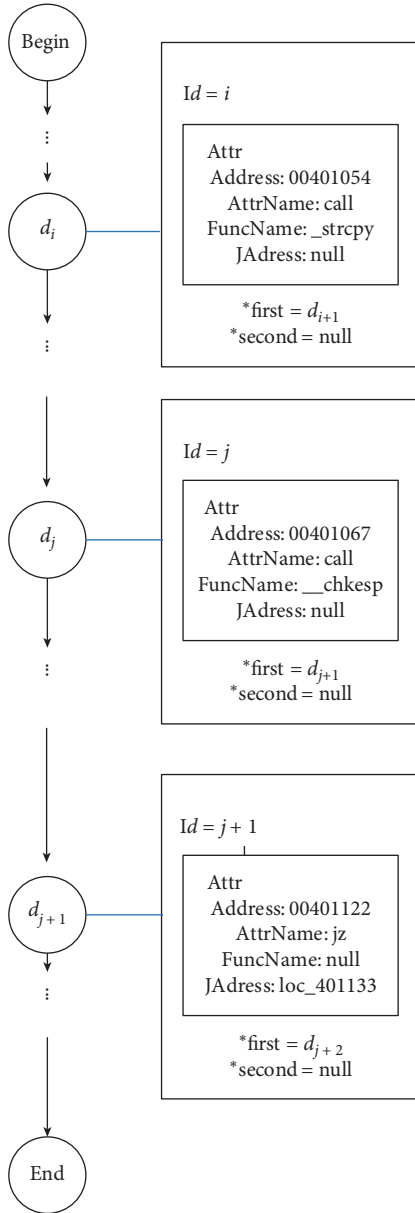


FIGURE 10: JCFG of the exploit.

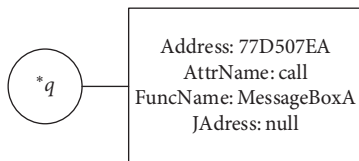


FIGURE 11: Program command node information during abnormal jump.

corresponding JCFG is generated and stored in the database. After obtaining the JCFG, the program is analyzed dynamically, reading the instruction information during each step-by-step execution, extracting information from the instruction information containing key nodes, and finally, obtaining the corresponding instruction node, calling the corresponding instruction determination function, and

```

Class Node
{
    String address; // Instruction address information
    String attrName; // Command name
    String funcName; // Instruction call function name
}

```

FIGURE 12: Implementation of node attribute data structure.

statistically analyzing the pointer nodes of the obtained JCFG with the node which obtain comparison to determine whether there is an abnormal jump, so as to determine whether the program to be detected belongs to an exploit.

5.2. Analysis of Results. The Vulnerability Exploit Detection Method based on JCFG can successfully detect the exploit shown in Table 1 and store the nodes that produce abnormal jumps, which are convenient for security personnel to analyze the exploit. First, the exploit is prevented by the protective measures in the system, and then the exploit is detected by the JCFG-VEDM method proposed in this paper. The detection results are shown in Table 2.

In Table 2, the detection result section uses “1” to represent that the detection method can detect the exploit and uses “0” to represent that the exploit cannot be detected. It can be seen from the results in Table 2 that neither the DEP protection strategy of the system nor the ASLR address randomization protection strategy can protect the system against the above-mentioned exploits. However, the Vulnerability Exploit Detection Method based on JCFG proposed in this paper can detect the above. It reflects that JCFG-VEDM can detect common ret-to-libc, ROP, and JIT Spraying vulnerabilities. Hence, the effectiveness and feasibility of the JCFG-VEDM detection method are verified.

Here, the detection process and detection results of the Vulnerability Exploit Detection Method based on JCFG are described.

CVE-2017-8869 is caused by a buffer overflow vulnerability in MediaCoder. Attackers can construct the .m3u file to cause the program buffer overflow and overwrite the return address of the program to execute arbitrary code. In this experiment, MediaCoder runs on the experimental host of windows 7, and the assembly code of the program is also monitored through Ollydbg. During operation, the node with ID 476 at 0x1400f92d6L shows that the successor node of this node should be the node with ID 477 in the database, but at runtime, the control flow jumps to another address. Thus, the instruction node does not match the program instruction node in the static JCFG in the generated program, and it is determined that an abnormal jump has occurred. The hacker can use it to execute the shellcode that hides in the .m3u file.

The experiment shows that the DEP and ASLR protection strategies fail to protect the system, the exploit program can execute the shellcode that hacker hid, and JCFG-VEDM can detect it by verity of the jump address of

TABLE 1: Vulnerability information about exploits.

CVE number	Vulnerability name	Software version
CVE-2018-9131	Reaper buffer error vulnerability	Reaper 5.78
CVE-2018-6481	Flexense Disk Savvy enterprise buffer error vulnerability	Flexense Disk Savvy enterprise 10.4.18
CVE-2017-14627	CyberLink LabelPrint buffer error vulnerability	CyberLink LabelPrint 2.5
CVE-2017-8869	MediaCoder buffer error vulnerability	MediaCoder 0.8.48.5888
CVE-2017-8870	AudioCoder buffer error vulnerability	AudioCoder 0.8.46

TABLE 2: The detection results of exploits.

CVE number	Detect method	Result
CVE-2018-9131	DEP	0
	ASLR	0
	JCFG-VEDM	1
CVE-2018-6481	DEP	0
	ASLR	0
	JCFG-VEDM	1
CVE-2017-14627	DEP	0
	ASLR	0
	JCFG-VEDM	1
CVE-2017-8869	DEP	0
	ASLR	0
	JCFG-VEDM	1
CVE-2017-8870	DEP	0
	ASLR	0
	JCFG-VEDM	1

executable file. And compared with the common CFI detection approach, JCFG-VEDM does not need the source code of the executable file to use the method of program instrumentation, it is convenient for security researchers to detect the vulnerability of executable program which cannot get the source code.

6. Threats to Validity

A threat to internal validity relates to the type of vulnerabilities used in the experimental analysis. To mitigate this threat, we have prepared more different CVE vulnerabilities regarding the buffer overflow vulnerability. A threat to external validity relates to the generalizability of our results because we used vulnerability data from only buffer overflow vulnerability to verify the effectiveness and the feasibility of the abnormal jump studied. Our future work will address this threat by examining other vulnerabilities like Heap Overflow, Stack Overflow, and so on.

7. Conclusion

There are certain features in the occurrence of program vulnerabilities. This paper conducts an in-depth analysis of the features of the exploits of abnormal jumps and proposes a Vulnerability Exploit Detection Method based on JCFG (JCFG-VEDM). Firstly, this method analyzes the exploit features of the exploit to obtain its corresponding JCFG. And then it uses the Ollydbg tool to dynamically analyze the exploit, generating corresponding instruction nodes for the executed key instructions. Finally, it compares nodes

pointed to by the JCFG node pointer to determine whether an abnormal jump has occurred. In addition, this method also can be utilized to determine whether the program is an exploit.

To verify the effectiveness and feasibility of the JCFG-VEDM method proposed in this paper, we compare the JCFG-VEDM with the current system's DEP protection strategy and ASLR address randomization protection strategy in the experimental analysis. Experimental results show that JCFG-VEDM can detect the above-mentioned exploits, while the system's DEP and ASLR protection strategies fail to protect the system, and compared with the traditional CFI, the JCFG-VEDM does not need instrumentation, and it is minimizing the workload for the security researchers to detect the exploits by using the approach.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was partly supported by the National Natural Science Foundation of China (NSFC) (Grant no. U1836116), the National Key R&D Program of China (Grant no. 2020YFB1005500), and the Leading-Edge Technology Program of Jiangsu Natural Science Foundation (Grant no. BK20202001).

References

- [1] N. R. Weidler, D. Brown, and S. A. Mitchell, "Return-oriented programming on a resource constrained device," *Sustainable Computing: Informatics and Systems*, vol. 22, pp. 244–256, 2019.
- [2] Y. H. Xu and Z. X. Sun, "Research development of abnormal traffic detection in software defined networking," *Journal of Software*, vol. 31, no. 1, pp. 183–207, 2020.
- [3] J. Qu, C. L. Fan, G. Y. Chen et al., "Research on establishment of network security service ability system for A new era," *Netinfo Security*, vol. 19, no. 1, pp. 83–87, 2019.
- [4] F. F. Wang, T. Zhang, W. G. Xu et al., "Overview of control-flow hijacking attack and defense techniques for process," *Chinese Journal of Network and Information Security*, vol. 5, no. 6, pp. 10–20, 2019.

- [5] M. H. Wang, H. Yi, A. V. Bhaskar et al., “Binary code continent: finer-grained control flow integrity for stripped binaries,” *Journal of Cyber Security*, vol. 1, no. 2, pp. 61–72, 2016.
- [6] B. Liu, J. F. Chen, S. L. Qin et al., “An approach based on the improved SVM algorithm for identifying malware in network traffic,” *Security and Communication Networks*, vol. 2021, Article ID 5518909, 14 pages, 2021.
- [7] M. Abadi, M. Budiu, Ú. Erlingsson et al., “Control-flow integrity principles, implementations, and applications,” *ACM Transactions on Information and System Security*, vol. 13, no. 1, pp. 1–40, 2009.
- [8] M. Abadi, “Protection in programming-language translations,” in *Proceedings of the 25th International Conference on Automata Languages and Programming (ICALP’98)*, pp. 868–883, Aalborg, Denmark, July 1998.
- [9] E. Göktas, E. Athanasopoulos, H. Bos et al., “Out of control overcoming control-flow integrity,” in *Proceedings of the 2014 IEEE Symposium International Conference on Security and Privacy*, pp. 575–589, San Jose, CA, USA, May 2014.
- [10] M. H. Wang, L. Y. Ying, and D. G. Feng, “Exploit detection based on illegal control flow transfers identification,” *Journal on Communications*, vol. 35, no. 9, pp. 20–31, 2014.
- [11] Ú. Erlingsson, M. Abadi, M. Vrable et al., “XFI: software guards for system Address spaces,” in *Proceedings of the 7th Symposium International Conference on Operating Systems Design and Implementation (OSDI’06)*, pp. 75–88, Seattle, WA, USA, November 2006.
- [12] R. D. Clercq, R. D. Keulenaer, B. Coppens et al., “SOFIA: software and control flow integrity architecture,” *Computers & Security*, vol. 68, pp. 16–35, 2017.
- [13] K. Heydemann, J. F. Lalande, and P. Berthomé, “Formally verified software countermeasures for control-flow integrity of smart card C code,” *Computers & Security*, vol. 85, pp. 200–218, 2019.
- [14] M. W. Zhang and R. Sekar, “Control flow and code integrity for COTS binaries,” in *Proceedings of the 22nd Symposium International Conference on USENIX Security (Usenix’13)*, pp. 337–352, Washington DC, USA, August 2013.
- [15] N. Dautenhahn, J. Criswell, and V. Adve, “KCoFI: complete control-flow integrity for commodity operating system kernels,” in *Proceedings of the 2014 IEEE Symposium International Conference on Security and Privacy*, pp. 292–307, San Jose, CA, USA, May 2014.
- [16] N. Carlin, A. Barresi, D. Wagner et al., “Control-flow bending: on the effectiveness of control-flow integrity,” in *Proceedings of the 24th Symposium International Conference on USENIX Security (Usenix’15)*, pp. 161–176, Washington DC, USA, August 2015.
- [17] C. Zhang, T. Wei, Z. F. Chen et al., “Practical control flow integrity and randomization for binary executables,” in *Proceedings of the 2013 IEEE symposium International Conference on Security and Privacy*, pp. 559–573, San Francisco, CA, USA, February 2013.
- [18] Y. B. Xia, Y. T. Liu, H. B. Chen et al., “CFIMon: detecting violation of control flow integrity using performance counters,” in *Proceedings of the 42nd Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN’12)*, pp. 1–12, Washington DC, USA, June 2012.
- [19] E. J. Schwartz, T. Avgerinos, and D. Brumley, “All you ever wanted to know about dynamic taint analysis and forward symbolic execution (but might have been afraid to ask),” in *Proceedings of the IEEE Symposium International Conference on Security and Privacy*, pp. 317–331, Oakland, CA, USA, May 2010.
- [20] X. J. Wang, C. Z. Hu, R. Ma et al., “A survey of the key technology of binary program vulnerability discovery,” *Netinfo Security*, vol. 17, no. 8, pp. 1–13, 2017.
- [21] Z. B. Han, X. H. Li, Z. C. Xing et al., “Learning to predict severity of software vulnerability using only vulnerability description,” in *Proceedings of the 2017 IEEE International Conference on Software Maintenance and Evolution (ICSME)*, pp. 125–136, Shanghai, China, September 2017.
- [22] J. X. Ma, Z. J. Li, T. Zhang et al., “Taint analysis method based on offline indices of instruction trace,” *Journal of Software*, vol. 28, no. 9, pp. 2388–2401, 2017.
- [23] L. Wang, F. Li, L. Li et al., “Principle and practice of taint analysis,” *Journal of Software*, vol. 28, no. 4, pp. 860–882, 2017.
- [24] M. V. Belyaev, N. V. Shimchik, V. N. Ignatyev et al., “Comparative analysis of two approaches to static taint analysis,” *Programming and Computer Software*, vol. 44, no. 6, pp. 459–466, 2018.
- [25] S. Sayeed, H. Marco-Gisbert, I. Ripoll et al., “Control-flow integrity: attacks and protections,” *Applied Sciences*, vol. 9, no. 20, p. 4229, 2019.
- [26] P. H. Yuan, Q. K. Zeng, Y. J. Zhang et al., “Attacking web browser: ROP gadget injection by using JavaScript code blocks,” *Journal of Software*, vol. 31, no. 2, pp. 247–265, 2020.

Research Article

Blockchain-Based Efficient Device Authentication Protocol for Medical Cyber-Physical Systems

Fulong Chen , Yuqing Tang , Xu Cheng , Dong Xie , Taochun Wang ,
and Chuanxin Zhao 

Anhui Normal University, Wuhu, China

Correspondence should be addressed to Fulong Chen; long005@mail.ahnu.edu.cn

Received 28 February 2021; Revised 5 April 2021; Accepted 16 April 2021; Published 4 May 2021

Academic Editor: Ke Gu

Copyright © 2021 Fulong Chen et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

As the background of application in the field of smart health care, the flexible interaction between patients and medical system is provided by medical cyber-physical systems (MCPSs) to realize all-round three-dimensional medical service. According to the controllable and credible requirements of MCPS, it needs a secure and reliable device identity authentication mechanism to build the security barrier. Based on the blockchain technology, a lightweight authentication scheme is designed for sensor/execution devices, users, and gateway nodes in MCPS. The security analysis and experimental results show that the scheme can resist the existing attacks with better efficiency; thus, our proposed scheme can be efficiently applied to the medical field.

1. Introduction

We have witnessed the great development of the Internet, as well as the popularity of the Internet of Things (IoT) and IoT devices, including wireless sensors, smart phones, wearable devices, global positioning systems, and laser scanners. These devices are widely deployed around us to realize intelligent computing and services, such as logistics, retail, medical, intelligent city, and other application fields. However, the trusted authentication in IoTs has become a major issue that has to be considered in the rapid development of IoTs.

Closely related to IoTs, medical cyber-physical systems (MCPSs) [1] are a kind of unique cyber-physical systems (CPSs) in the field of modern medicine, which combines the system operations with independent equipment to provide patients with new monitoring functions, such as controlling the physiological closed loop and alarm process of drug infusion process. In the MCPS, there are many kinds of devices with different performance. With the development of blockchain technology, the blockchain-based

authentication schemes can mitigate some attacks, which ensure the security of the system. How to ensure that the security authentication protocol can work efficiently and reliably when using the blockchain technology is the key problem to be solved. Hence, based on the blockchain technology, we propose a device authentication scheme to ensure secure access to medical data among sensor devices nodes, gateway nodes, and users in the medical cyber-physical system. Specifically, our contributions can be summarized as follows:

- (1) We distinguish the identity of the device nodes in the information physical space and propose a device security authentication model based on blockchain for the medical cyber-physical system.
- (2) We design a blockchain-based efficient device authentication protocol. Our scheme is suitable for device nodes with different computing, transmission, and storage capacities and uses blockchain technology to solve the trustworthiness problem of third-party service centers. Meanwhile, we use BAN

logic and formal proof to verify the feasibility of our scheme and the security of mutual authentication process and the session key.

2. Related Works

Based on the extensive application of radio frequency identification (RFID) in medical environment, He et al. [2] analyzed the security requirements of RFID authentication scheme and summarized the performance and security of RFID authentication scheme based on elliptic curve cryptography (ECC). They found that although most authentication schemes cannot meet all the security requirements and have satisfactory performance, some ECC-based authentication schemes are suitable for medical environment in terms of performance and security. Combined with cloud storage, cryptography, and other technologies, a large number of authentication schemes are also proposed. The wireless body area network (WBAN) plays an indispensable role in MCPS. It is a network composed of multiple wearable devices or embedded devices, using wireless technology for communication. Therefore, in WBAN environment, a security and reliable authentication scheme is essential. Xu et al. [3] proposed a safe lightweight authentication scheme for WBAN. With this scheme, forward secrecy can be guaranteed without asymmetric encryption, and the security of the scheme can be verified and analyzed by using ProVerif. Alhayajneh et al. [4] analyzed and evaluated the accuracy, cost, and feasibility of the most prominent biometric authentication technology and proposed to use a variety of biometric authentication schemes to ensure the confidentiality, integrity, and reliability of WBAN. Moosavi et al. [5] proposed an end-to-end security scheme for mobile medical IoTs. Their solutions include a secure and efficient end-user authentication and authorization architecture based on certificate DTLS handshake, end-to-end communication based on session recovery security, and strong mobility based on Internet intelligent gateway. Amin et al. [6] proposed a mutual authentication and key agreement protocol to protect the confidential information in the device in order to prevent unauthorized users from accessing the general device. Aiming at the challenges brought by the electronic health information management system using IoTs, including the communication security of wireless channel, the protocol between authentication key and entity, access control scheme, and other defects, Aghili et al. [7] proposed a new lightweight, secure, and efficient authentication protocol, which is also suitable for access control. Aiming at the problem of authentication in edge and IoT environments, Ma et al. [8] proposed a blockchain-based decentralized authentication modeling scheme. Their scheme is suitable for multiple types of authentication (such as password-based, certificate-based, biometric-based, and token-based authentication). The edge cloud system also has many devices with limited computing and storage capabilities. Thus, Zhang et al. [9] proposed a collaborative authentication scheme among users, edge cloud, and robots, which reduced the computational cost of identity verification and improved the verification efficiency. In order to

provide more accurate and effective biometric identification, Zhang et al. [10] proposed a parallel ECG-based authentication called PEA for smart healthcare systems.

According to the survey of Altman Vilandrie and Company [11], due to the lack of security authentication and other security systems, the IoT system of small- and medium-sized enterprises is vulnerable to attacks, resulting in their annual income loss of up to 13%. Chandrasekhar et al. [12] reported that the protocol of Yeh's protocol has some shortcomings, including incomplete forward secrecy, non-mutual authentication, and key agreement between users and sensor nodes. Shi and Gong [13] proposed an ECC-based user authentication protocol for wireless sensor networks, which is more efficient in computing cost, communication cost and security. However, Choi et al. [14] found that the protocol of Shi is vulnerable to session key attack, stolen smart card attack, and sensor energy depletion attack. In addition, an attacker can easily obtain the user's identity because it is transmitted through a public channel without encryption. Therefore, Choi et al. improved the protocol by verifying the identification legitimacy of users so as to keep from sensor energy consumption attacks. Compared with the protocol of Shi, the protocol also makes use of ECC to calculate authentication messages without bringing more cost. Both [13, 14] transmit user identity and sensor identity in plaintext on the public channel, so that they cannot provide anonymity. Chen et al. [15] proposed transmission protection, storage protection, and access control of infrastructure framework in the context of privacy protection of community medical IoT but did not mention the device security authentication. Shu et al. [16] proposed the aggregate signature algorithm, but it lacks the application background. Xue et al. [17] proposed a wireless sensor network identity authentication and key protocol based on temporary credentials, which only uses hash and XOR calculation. It has relatively more security features and higher security level without generating more communication and computing costs, but the traditional third party is vulnerable to attack. In order to solve the problem that the restricted computing power and storage of the sensors are vulnerable to physical attacks, Liu et al. [18] proposed a lightweight three-factor and anonymous user authentication protocol. The solution uses hash algorithms, XOR operations, and PUF to achieve lightweight and physical security. The wireless sensor network is widely used in medical, military, industrial, security, and other fields. Recently, Kumar et al. [19] discussed a wireless sensor network authentication protocol for coal mine safety monitoring. In the IoT environment, trust has become ubiquitous. It is not enough to just authenticate individual users or devices. The reason is that the cointeraction and cooperation between users and devices are crucial in the IoT environment. In this case, information sharing, data fusion, and other elements, including the integration of people, devices, and environment, are great challenges.

Traditional device authentication methods usually perform authentication when users and devices are separated from each other. At the same time, attackers can eavesdrop on communications, forge authentication tokens [20], or

perform replay attacks to simulate actual users or devices. The existing authentication schemes rarely consider the space-time characteristics of IoT computing. In general, authentication usually performs settings at once, and once users or devices are authenticated, they can operate for a long time without any authentication. This kind of time-sensitive authentication still needs to be improved so as to realize sufficient long-term trust guarantee, which is continuous authentication. Once an attacker fortunately bypasses the authentication system, various destructive attacks can be carried out. Therefore, the system can only respond to the attack passively, and the security network of IoTs is threatened.

Recently, researchers have gradually applied blockchain to the medical field. The combination of MCPS and blockchain can allow us to promote the sharing of services and resources and simplify several time-consuming workflows in an automated manner during the encryption verification [21]. In cloud-assisted telecare medical information system (TMIS), cloud servers are vulnerable to attacks. To solve this problem, Son et al. [22] used blockchain technology to design a secure identity verification protocol. In addition, they used CP-ABE to achieve data access control. Although the blockchain-based authentication scheme can enhance the security of the system, it is necessary to consider the authentication credentials and the accounting method of the authentication process when we use the decentralized blockchain as a third party to achieve authentication. Especially, the existing blockchain consensus algorithms and authentication protocols no longer adapt to the wide range of devices with vastly different performance in MCPS.

3. MCPS and Its Security Model

In the part of related works, we analyze the existing security risks and threats of device authentication. Therefore, we need to further improve the device authentication scheme to ensure the safety and reliability.

3.1. Classification of Medical Devices. With the rapid and revolutionary development of medical information, the medical equipment is widely used. The medical devices are classified as follows.

The first kind is the diagnostic equipment. It includes physical diagnostic instruments (sphygmomanometers, thermometers, all kinds of physiological recorders, etc.), images (MRI, B ultrasound, CT scanning, etc.), analytical instruments, and electrophysiology (EEG, etc.). These devices are distributed in each diagnosis and treatment area of the hospital, and the devices connected to the network need strict identity authentication.

The second kind is the treatment equipment. It includes ward nursing equipment (sickbed, oxygen bottle, etc.), surgical equipment, radiotherapy equipment, and emergency equipment (ventilator, cardiac defibrillation pacemaker, etc.). This kind of devices needs to be authenticated to ensure the safe use.

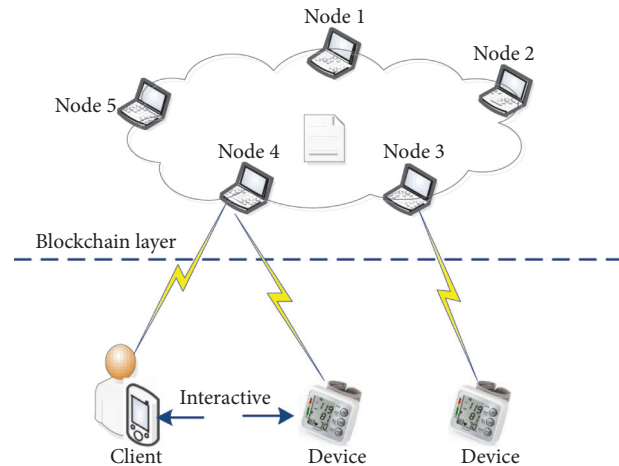


FIGURE 1: Model of device security authentication.

The third kind is the auxiliary equipment. It includes sterilization devices, refrigeration devices, and so on.

3.2. System Model. In order to study the problem of device security authentication in the MCPS, we first construct the system model of device security authentication based on the blockchain, as shown in Figure 1. The medical institutions are organized in a medical alliance chain to realize medical data sharing. The lower layer of the blockchain is composed of users and some medical equipment, which mainly completes data collection and other work; the blockchain layer is mainly used to realize the storage of medical data and the process of device security authentication.

As shown Figure 1, the device is mainly composed of sensor nodes. Sensor nodes can perceive various characteristics from different environments. In the MCPS, the collection process of medical data is mainly composed of medical professionals (doctors, patients, nurses, pathologists, etc.), sensors, and gateway nodes, as shown in Figure 2. The sensor nodes sense the patient's physical condition and then send the sign data in a certain electronic data format to the trusted gateway nodes of MCPS through the access point. As the core of the model in MCPS, the trusted gateway nodes execute the registration algorithm to provide the registration interface to all medical staffs. Medical staffs collect sensitive sign information of patients from the trusted gateway nodes, analyze them, and monitor patients' physical condition.

3.3. Architecture. MCPS increasingly relies on software to provide new functions, so that new medical software and devices can be more widely connected with the network to meet the needs of continuous monitoring of patients. The basic architecture of MCPS includes cyber space (including network space) and physical space (including user space), as shown in Figure 3. As the core of MCPS, cyber space includes the processing, storage, security access, and so on. Physical space is the physical basis of MCPS, including medical perception and control devices needed by users, such as electronic sphygmomanometer, heart rate, and pulse

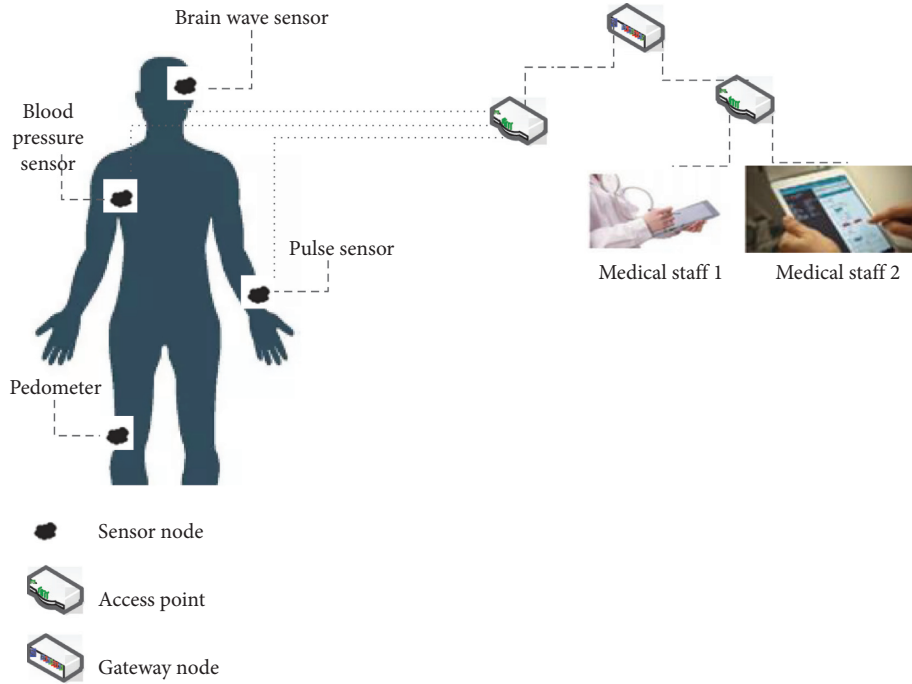


FIGURE 2: Collection process of medical data.

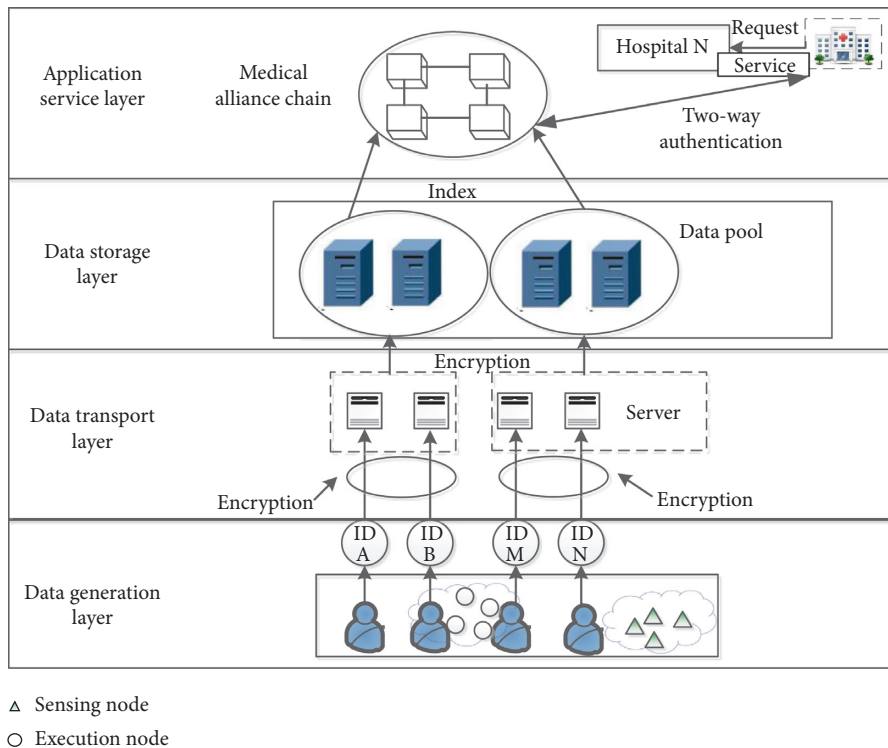


FIGURE 3: Blockchain-based architecture of MCPS.

collector, which are responsible for the collection and monitoring of user health information. MCPS is composed four layers such as data generation layer, data transport layer, data storage layer, and application service layer.

3.3.1. *Data Generation Layer.* At the bottom of MCPS architecture, it is mainly composed of a series of sensing nodes to collect the user's health information and transmit the collected data to the medical data storage space through the

tablet or other electronic devices. At the same time, after receiving the feedback information from the sensing nodes, the execution nodes complete the monitoring function of the user through the display or other alarms and execution devices and timely transmit the received information to the outside world, so as to realize the communication and feedback of information.

3.3.2. Data Transport Layer. It means that after the data collected by the data generation layer is encrypted or processed by other means, it is transmitted to the server by wired Ethernet or wireless transmission for storage. The service of the data transport layer is reflected through the running IPv4 or IPv6 protocol. Therefore, the level of data transport layer has a great relationship with the quality of network service, which requires higher network bandwidth, larger transmission range, faster transmission rate, stable transmission process performance, etc.

3.3.3. Data Storage Layer. The data storage object is the user's health information. The health information includes the user's medical records (sign data, outpatient medical records, hospitalization records, body temperature list, doctor's order list, laboratory test list, medical imaging examination data, special examination consent, operation consent, operation and anesthesia record list, pathological data, nursing records, and other medical records), etc. It uses blockchain and cloud storage technology to realize the secure storage and data sharing of medical records. The chain structure that stores the hash of medical data of each hospital, the digest, and the location index of medical data in cloud storage is called medical chain.

3.3.4. Application Service Layer. It focuses on application management and secure access to medical data. With the help of the platform technology of blockchain, we can provide data addition, deletion, insertion, decision-making, and diversified services. And the user can send the corresponding operation or control commands to the relevant execution nodes according to the access results to realize the feedback and exchange of information.

In the system model, we mainly combine the blockchain technology, build a system model of security authentication, and analyze the collection process of medical data under the blockchain. On the basis of this model, we need to consider how to ensure the security of the device node access. Therefore, we propose a secure data transmission protocol based on device authentication and key agreement. The proposed authentication protocol consists of six stages: system setup, user registration, user login, authentication, key change, and sensor node join. The symbols and descriptions used are shown in Table 1.

TABLE 1: Symbols and definitions.

Symbol	Description
BC	Blockchain center
SC	Smart card
U_i	User i (medical staff)
ID_i	ID of The user U_i
PW_i	Password of user U_i
GW_j	Gateway j
ID_g	Gateway node identifier
SN_k^g	The K^{th} sensor device node
ID_{sn}	Identifier of the sensor device node
S_{sn}	Shared key of sensor device and gateway node
SK_i	Session key
\parallel	Connection operation
\oplus	Exclusive or operation

4. Authentication Protocol

In this section, we propose a device security authentication scheme based on blockchain technology to ensure the security and reliability of sensor device nodes. The proposal in this section mainly includes the following parts.

4.1. Setup

- (i) Step 1: the blockchain center BC selects S_{BC} as its private key and ID_g as the identifier of the gateway node and calculates

$$S_g = h(ID_g \parallel S_{BC}), \quad (1)$$

where S_g is selected as the private key of the gateway node.

- (ii) Step 2: ID_{sn} is the identifier of the sensor device node selected by the blockchain center BC, which calculates

$$S_{sn} = h(ID_{sn} \parallel S_{BC}). \quad (2)$$

It is the shared key between the gateway node and the sensor device node.

- (iii) Step 3: $\{ID_{sn}, S_{sn}\}$ is saved in SN_k by the blockchain center BC.
- (iv) Step 4: the blockchain center BC saves $\{ID_g, S_g, ID_{sn}, S_{sn}\}$ and sends it to the gateway node GW_j in order to register SN_k with the gateway node.

4.2. User Registration. In order to access the medical data collected from sensor device nodes, each medical staff needs to register at the corresponding gateway. In the user registration stage, the medical staff sends the registration request to the gateway. After the preliminary verification, the gateway adds the user into its user list and sends one smart card storing user's identification information to the user. The

identification information may include some personalized parameters of the user, such as complex password in a certain length and identity credentials convenient for authentication in encrypted form. The steps of user registration are as follows:

- (i) Step 1: the user selects a unique ID_i and PW_i , generates a random number r_1 , and calculates

$$HPW_i = h(r_1 \oplus PW_i). \quad (3)$$

Then, the user sends $\{ID_i, HPW_i\}$ to the gateway node GW_j .

- (ii) Step 2: when the gateway node GW_j receives $\{ID_i, HPW_i\}$, the gateway node GW_j generates another random number r_2 and calculates it at the timestamp T_1 :

$$\begin{aligned} R_1 &= h(HPW_i \| T_1), \\ R_2 &= h(HPW_i \| ID_g), \\ R_3 &= h(R_1 \| r_2 \| S_g) \oplus h(HPW_i \| T_1). \end{aligned} \quad (4)$$

- (iii) Step 3: the gateway node GW_j stores $\{r_2, T_1, ID_g, h(\cdot), R_1, R_2, R_3\}$ in the smart card SC and then transmits it to the user U_i securely.
- (iv) Step 4: when the user U_i receives $\{r_2, T_1, ID_g, h(\cdot), R_1, R_2, R_3\}$, U_i calculates

$$HID = h(PW_i \| ID_i) \oplus r_1. \quad (5)$$

And it writes it into the smart card.

4.3. User Login. In the login stage, the user U_i enters the identity identifier (identity credentials and password) in the device. The system first checks the correctness of the user input value and then sends the login message to the gateway. Once the authentication is successful, the user U_i can securely and legally access the remote computer data at any time according to the following steps:

- (i) Step 1: the user U_i inserts the SC into the reader, then enters ID_i and PW_i .
- (ii) Step 2: the user U_i selects a gateway node GID_j to obtain the data required by the user from the nearest sensor node.
- (iii) Step 3: smart card calculates

$$\begin{aligned} r_1^* &= HID \oplus h(PW_i \| ID_i), \\ HPW_i^* &= h(r_1^* \oplus PW_i), \\ R_2^* &= h(HPW_i^* \| ID_g). \end{aligned} \quad (6)$$

- (iv) Step 4: the smart card checks whether R_2^* and R_2 are equal. If $R_2 = R_2^*$, the ID_i and PW_i of the user are verified; otherwise, the session is interrupted.
- (v) Step 5: the smart card generates a random number r_3 , and at T_2 , it calculates

$$\begin{aligned} F_1 &= R_3 \oplus h(HPW_i \| T_1), \\ F_2 &= h(T_2 \| r_3 \| F_1 \| ID_g), \\ F_3 &= (r_3 \| T_2) \oplus F_1. \end{aligned} \quad (7)$$

- (vi) Step 6: the smart card sends $\{ID_{sn}, F_2, F_3\}$ to the gateway node GW_j through the public channel.

4.4. Authentication. In the authentication stage, the gateway node first verifies the validity of the user's identity and then transmits the authentication message to the sensor device. After receiving the authentication message, the sensor device verifies the identification authenticity of the gateway node and then sends another message back to the gateway node so as to further prove its authenticity. After that, the gateway node sends a new message to the user node. In addition, the session key is calculated by each participant, including user nodes, gateway nodes, and sensor device nodes. In this stage, the following steps are performed to establish mutual authentication between the caregiver user node and the sensor device node.

- (i) Step 1: when gateway node GW_j receives the login request $\{ID_{sn}, F_2, F_3\}$ at time T_3 , GW_j calculates

$$F_1^* = R_3 \oplus h(HPW_i \| T_1) = h(R_1 \| r_2 \| S_g), \quad (8)$$

$$F_1^* \oplus F_3 = (r_3^* \| T_2^*).$$

- (ii) Step 2: the gateway node GW_j checks whether $(T_3 - T_2)$ is less than ΔT , where ΔT is the maximum allowable transmission delay of the sender and receiver. If the condition is not met, terminate the session; otherwise, continue to the next step.
- (iii) Step 3: the gateway node GW_j calculates

$$F_2^* = h(T_2^* \| r_3^* \| F_1^* \| ID_g). \quad (9)$$

And it checks whether $F_2^* = F_2$. If met, the user U_i is authenticated; otherwise, the session is terminated.

- (iv) Step 4: the gateway node GW_j generates a random number r_4 and calculates

$$\begin{aligned} R_4 &= h(ID_{sn} \| R_1 \| S_{sn} \| r_4 \| T_3), \\ R_5 &= (r_3^* \| T_3 \| r_4) \oplus S_{sn}, \\ R_6 &= R_1 \oplus h(ID_{sn} \| h(r_4) \| r_3^*). \end{aligned} \quad (10)$$

- (v) Step 5: gateway node GW_j sends $\{ID_{sn}, R_4, R_5, R_6\}$ to sensor node SN_k .
- (vi) Step 6: after the SN_k receives $\{ID_{sn}, R_4, R_5, R_6\}$, then at time T_4 , it calculates

$$(r_3^{**} \| r_4^* \| T_3^*) = R_5 \oplus S_{sn}. \quad (11)$$

- (vii) Step 7: sensor device node SN_k checks whether $(T_4 - T_3)$ is less than ΔT . If the condition is not

met, terminate the session; otherwise, continue to the next step.

(viii) Step 8: sensor device node SN_k calculates

$$\begin{aligned} R_1^* &= R_6 \oplus h(\text{ID}_{\text{sn}} \| h(r_4^*) \| r_3^{**}), \\ R_4^* &= h(\text{ID}_{\text{sn}} \| R_1^* \| S_{\text{sn}} \| r_4^* \| T_3^*). \end{aligned} \quad (12)$$

(ix) Step 9: the SN_k checks whether $R_4^* = R_4$. If met, it continues to the next step; otherwise, terminate the session.

(x) Step 10: the SN_k generates a random number r_5 and calculates

$$\begin{aligned} SK_i &= h(R_1^* \| r_3^{**} \| r_4^* \| r_5), \\ B_1 &= h(T_4 \| r_5 \| S_{\text{sn}} \| \text{ID}_{\text{sn}} \| T_3 \| SK_i), \\ B_2 &= h(r_5 \| T_4) \oplus r_4^*. \end{aligned} \quad (13)$$

(xi) Step 11: the SN_k sends $\{B_1, B_2\}$ to the gateway node GW_j .

(xii) Step 12: the GW_j receives the message $\{B_1, B_2\}$ at time T_5 and calculates

$$(r_5^* \| T_4^*) = B_2 \oplus r_4. \quad (14)$$

And it checks whether $(T_5 - T_4)$ is less than ΔT . If not met, the session is terminated; otherwise, continue to the next step.

(xiii) Step 13: the GW_j verifies whether $B_1^* = B_1$; if met, the SN_k is verified.

(xiv) Step 14: the GW_j continues to calculate

$$\begin{aligned} R_7 &= h(SK_i \| R_1 \| r_4 \| T_5 \| R_4), \\ R_8 &= (r_5^* \| r_4 \| T_5) \oplus r_3^*. \end{aligned} \quad (15)$$

(xv) Step 15: the GW_j sends $\{R_4, R_7, R_8\}$ to the user U_i .

(xvi) Step 16: when the user receives $\{R_4, R_7, R_8\}$ at time T_6 , the smart card calculates

$$(r_5^{**} \| r_4^* \| T_5^*) = R_8 \oplus r_3. \quad (16)$$

And it checks whether $(T_6 - T_5) \leq \Delta T$; if not met, then terminate the session; otherwise, continue to the next step.

(xvii) Step 17: smart card calculates

$$\begin{aligned} R_7^* &= h(SK_i \| R_1 \| r_4^* \| T_5^* \| R_4) \\ &= h(SK_i \| h(\text{HPW}_i \| T_1) \| r_4^* \| T_5^* \| R_4). \end{aligned} \quad (17)$$

If $R_7^* = R_7$, then both GW_j and SN_k authenticate with user U_i ; otherwise, the session is terminated.

Among them, security authentication and key agreement phase is shown in Figure 4.

4.5. Password Change. This stage provides the user with the operation to change the password. An effective password change process can make the protocol friendly. In order to achieve this goal, the password change should not involve any other unnecessary participants, which can reduce communication costs and resist Denial of Service (DoS) attacks. Here are the steps to change the password:

(i) Step 1: the user U_i first inserts the SC into the reader device and then enters ID_i and PW_i .

(ii) Step 2: SC calculates

$$\begin{aligned} r_1^* &= \text{HID} \oplus h(\text{PW}_i \| \text{ID}_i), \\ \text{HPW}_i^* &= h(r_1^* \oplus \text{PW}_i), \\ R_2^* &= h(\text{HPW}_i^* \| \text{ID}_g), \\ h(R_1 \| r_2 \| S_g) &= R_3 \oplus (\text{HPW}_i \| T_1). \end{aligned} \quad (18)$$

(iii) Step 3: SC compares R_2^* with R_2 already stored in SC. If $R_2^* = R_2$, the user identity and password are verified; otherwise, the session is terminated.

(iv) Step 4: U_i enters a new password PW_i^{new} .

(v) Step 5: SC calculates

$$\begin{aligned} \text{HID}^{\text{new}} &= r_1^* \oplus h(\text{PW}_i^{\text{new}} \| \text{ID}_i), \\ \text{HPW}_i^{\text{new}} &= h(r_1^* \oplus \text{PW}_i^{\text{new}}), \\ R_2^{\text{new}} &= h(\text{HPW}_i^{\text{new}} \| \text{ID}_g), \\ R_3^{\text{new}} &= h(R_1 \| r_2 \| S_g) \oplus h(\text{PW}_i^{\text{new}} \| T_1). \end{aligned} \quad (19)$$

(vi) Step 6: the SC replaces R_2, R_3 and HID with the corresponding new values: $R_2^{\text{new}}, R_3^{\text{new}}$, and HID^{new} . Then, the password is changed successfully.

4.6. Sensor Node Join. When a new sensor device node needs to join the MCPS, the system will perform the following steps:

(i) Step 1: the blockchain center BC selects the new sensor node SN_k , uses ID_{sn} as its identifier, and calculates

$$S_{\text{sn}} = h(\text{ID}_{\text{sn}} \| S_{\text{BC}}). \quad (20)$$

And it stores $\{SN_k, S_{\text{sn}}\}$.

(ii) Step 2: BC sends $\{SN_k, S_{\text{sn}}\}$ to the gateway node.

(iii) Step 3: the gateway node stores this value and updates the information in the database.

5. Authentication Proof Based on Ban Logic

5.1. Definition of BAN. BAN (Burrows, Abadi, and Needham) logic is a popular identity authentication protocol analysis model. It helps to prove identity verification and key

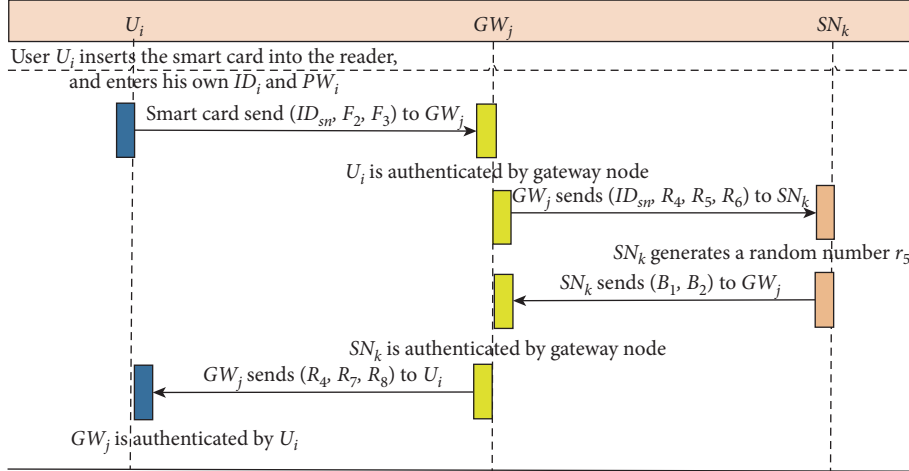


FIGURE 4: Security authentication and key agreement phase.

establishment, thereby proving the validity of the protocol [23, 24].

Now, we have defined some logical rules such as Message-Meaning rule (MM), Nonce-Verification rule (NV), Jurisdiction rule (J), Freshness-Conjunction rule (FC), Session Key rule (SK), and Belief rule (B) used in the proof, and these rules are directly adopted from [25].

5.2. Formal Verification Process. All certification protocols need to achieve Goal 1, 2, ..., 8. Here, the variables U_i , GW_j , and SN_k represent three subjects:

- (i) Goal 1: $GW_j | \equiv U_i \stackrel{SK_i}{\leftrightarrow} GW_j$.
- (ii) Goal 2: $GW_j | \equiv U_i \stackrel{SK_i}{\equiv} U_i \stackrel{SK_i}{\leftrightarrow} GW_j$.
- (iii) Goal 3: $SN_k | \equiv GW_j \stackrel{SK_i}{\leftrightarrow} SN_k$.
- (iv) Goal 4: $SN_k | \equiv GW_j \stackrel{SK_i}{\equiv} GW_j \stackrel{SK_i}{\leftrightarrow} SN_k$.
- (v) Goal 5: $GW_j | \equiv SN_k \stackrel{SK_i}{\leftrightarrow} GW_j \stackrel{SK_i}{\leftrightarrow} SN_k$.
- (vi) Goal 6: $GW_j | \equiv SN_k | \equiv SN_k \stackrel{SK_i}{\leftrightarrow} GW_j$.
- (vii) Goal 7: $U_i | \equiv GW_j \stackrel{SK_i}{\leftrightarrow} U_i$.
- (viii) Goal 8: $U_i | \equiv GW_j | \equiv GW_j \stackrel{SK_i}{\leftrightarrow} U_i$.

Make the following assumptions and analyze the initial state of the agreement:

- (i) $A_1: U_i | \equiv \#(r_3, r_4, r_5)$.
- (ii) $A_2: GW_j | \equiv \#(r_3, r_4, r_5)$.
- (iii) $A_3: SN_k | \equiv \#(r_3, r_4, r_5)$.
- (iv) $A_4: U_i | \equiv U_i \stackrel{F_1}{\leftrightarrow} GW_j$.
- (v) $A_5: GW_j | \equiv GW_j \stackrel{S_{sn}}{\leftrightarrow} GW_j$.
- (vi) $A_6: SN_k | \equiv SN_k \stackrel{S_{sn}}{\leftrightarrow} GW_j$.
- (vii) $A_7: GW_j | \equiv GW_j \stackrel{F_1}{\leftrightarrow} U_i$.
- (viii) $A_8: GW_j | \equiv U_i = > r_3$.
- (ix) $A_9: SN_k | \equiv GW_j = > r_4$.
- (x) $A_{10}: GW_j | \equiv SN_k = > r_5$.
- (xi) $A_{11}: U_i | \equiv GW_j = > r_4$.

Based on BAN logic rules and assumptions, we can analyze the ideal form of the protocol:

- (i) Message 1: $U_i \longrightarrow GW_j: \langle ID_{sn}, F_2, F_3 \rangle$

Using $P \triangleleft X$ rule:

$$R_1: GW_j \triangleleft \langle ID_{sn}, F_2: (T_2, r_3)_{F_1}, F_3: \langle r_3, T_2 \rangle_{F_1} \rangle$$

Using A_7, R_1 and MM rule:

$$R_2: GW_j | \equiv U_i | \sim (T_2, r_3, ID_g)$$

Using A_1, R_2 and FC rule:

$$R_3: GW_j | \equiv U_i \equiv (T_2, r_3, ID_g)$$

Using A_8, R_3 and J rule, B rule, NV rule:

$$R_4: GW_j | \equiv r_3$$

Using A_2, R_4 and SK rule:

$$R_5: GW_j | \equiv U_i \stackrel{SK_i}{\leftrightarrow} GW_j \text{ (Goal 1)}$$

Using A_2, R_5 and NV rule:

$$R_6: GW_j | \equiv U_i | \equiv U_i \stackrel{SK_i}{\leftrightarrow} GW_j \text{ (Goal 2)}$$

- (ii) Message 2: $GW_j \longrightarrow SN_k: \langle ID_{sn}, R_4, R_5, R_6 \rangle$

Using $P \triangleleft X$ rule:

$$R_7: \langle \langle ID_{sn}, R_4, R_5: \langle r_3, r_4, T_3 \rangle_{S_{sn}}, R_6 \rangle \rangle$$

Using A_6, R_7 and MM rule:

$$R_8: SN_k | \equiv GW_j | \sim (r_3, r_4, T_3)$$

Using A_2, R_8 and NV rule:

$$R_9: SN_k | \equiv GW_j | \equiv (r_3, r_4, T_3)$$

Using A_2, R_9 and J rule, FC rule:

$$R_{10}: SN_k | \equiv (r_3, r_4, T_3)$$

Using R_{10} and B rule:

$$R_{11}: SN_k | \equiv r_4, SN_k | \equiv r_3$$

Using A_3, R_{11} and SK rule:

$$R_{12}: SN_k | \equiv GW_j \stackrel{SK_i}{\leftrightarrow} SN_k \text{ (Goal 3)}$$

Using A_3, R_{12} and NV rule:

$$R_{13}: SN_k | \equiv GW_j | \equiv GW_j \stackrel{SK_i}{\leftrightarrow} SN_k \text{ (Goal 4)}$$

- (iii) Message 3: $SN_k \longrightarrow GW_j: \langle B_1, B_2 \rangle$

Using $P \triangleleft X$ rule:

$R_{14}: GW_j \triangleleft \langle \langle B_1: (T_4, r_5, S_{sn}, ID_{sn}, T_3^*, SK_i)_{S_{sn}}, B_2 \rangle \rangle$

Using A_5, R_{14} and MM rule:

$R_{15}: GW_j | \equiv SN_k | \sim (T_4, r_5, SK_i)$

Using A_2, R_{15} and J rule, FC rule, NV rule:

$R_{16}: GW_j | \equiv SN_k | \equiv (T_4, r_5, SK_i)$

Using R_{16} and B rule:

$R_{17}: GW_j | \equiv r_5$

Using A_2, R_{17} and SK rule:

$R_{18}: GW_j | \equiv SN_k \xleftrightarrow{SK_i} GW_j$ (Goal 5).

Using A_2, R_{18} and NV rule:

$R_{19}: GW_j | \equiv SN_k | \equiv SN_k \xleftrightarrow{SK_i} GW_j$ (Goal 6).

(iv) Message 4: $GW_j \longrightarrow U_i: \langle R_4, R_7, R_8 \rangle$

Using $P \triangleleft X$ rule:

$R_{20}: U_i \triangleleft \langle \langle R_4, R_7, R_8: \langle T_5, r_4, r_5 \rangle_{r_3} \rangle \rangle$

Using A_4, R_4, R_{20} and MM rule:

$R_{21}: U_i | \equiv GW_j | \sim (T_5, r_4, r_5)$

Using A_1, R_{21} and FC rule, NV rule:

$R_{22}: U_i | \equiv GW_j | \equiv (T_5, r_4, r_5)$

Using A_{11}, R_{22} , B rule and J rule:

$R_{23}: U_i | \equiv r_4$

Using A_1, R_{23} and SK rule:

$R_{24}: U_i | \equiv GW_j \xleftrightarrow{SK_i} U_i$ (Goal 7).

Using A_1, R_{24} and NV rule:

$R_{25}: U_i | \equiv GW_j | \equiv GW_j \xleftrightarrow{SK_i} U_i$ (Goal 8).

The above BAN logic discussion clearly proves the effectiveness and feasibility of the mutual authentication and session key protocol among user U_i , gateway node GW_j , and sensor device node SN_k .

6. Security Analysis and Discussion

6.1. Security Analysis. In this section, we mainly discuss the security issues to prove that our protocol is secure for all related security attacks.

6.1.1. Replay Attack. Assuming that the device authentication protocol maintains a global clock to synchronize timestamps against clock synchronization, we can verify whether it can effectively resist replay attacks and work smoothly or not. Affected by replay attack, the performance of the system will decline dramatically. Attackers usually capture the previously transmitted messages by the sender entity and resend them to the receiver entity to prove that the message was sent from the legitimate sender entity. Because the system timestamp is used in the protocol and the transmission delay time ΔT will be checked, the protocol always rejects the replay messages captured by the attacker due to the invalid transmission delay time. In the protocol, new random numbers are also used to identify duplicate messages. Therefore, the protocol proposed in this paper is resistant to replay attacks.

6.1.2. User Impersonation Attack. According to the attacker's ability, the attacker can eavesdrop all the transmitted messages through the public channel during the execution of the protocol. The attacker can modify the bugged message and retransmit it to the user in order to impersonate a valid user. The following will prove that the protocol in this paper provides strong security protection against user simulated attacks.

We suppose that the attacker eavesdrops on the message $\{ID_{sn}, F_2, F_3\}$ and tries to generate another valid message, which will be authenticated by the gateway. In order to generate a forged message, the attacker must calculate the following valid parameters:

$$\begin{aligned} F_2 &= h(T_2 \| r_3 \| F_1 \| ID_g), \\ F_3 &= (r_3 \| T_2) \oplus F_1. \end{aligned} \quad (21)$$

However, the attacker could not calculate the effective $F_1 = R_3 \oplus h(HPW_i \| T_1)$, where $HPW_i = h(r_1 \oplus PW_i)$ as PW_i and r_1 are unknown to the attacker. In addition, it is not feasible to simulate and guess all unknown constraints in polynomial time. As a result, attackers cannot generate or guess other valid messages in polynomial time.

6.1.3. Offline User Identity and Password Guessing Attacks. Assuming that most users use simple ID_i and PW_i for identity recognition, it is easy to guess ID_i and PW_i in polynomial time. However, during the execution of the protocol in this paper, the user's ID_i and PW_i are protected by an irreversible one-way hash function. Therefore, the attacker cannot extract user information $\{ID_i, PW_i\}$. An attacker may try to extract multiple parameters such as $R_2, R_3, F_2, F_3, R_4, R_5, B_1$, and B_2 from the offline state of the user and then guess and verify user's ID_i and PW_i . All these parameters are known to the attacker as follows.

The attacker finds the constraint parameters F_2 and F_3 of the smart card. The constraint parameters F_2 and F_3 of the smart card are defined as

$$\begin{aligned} F_2 &= h(T_2 \| r_3 \| F_1 \| ID_g), \\ F_3 &= (r_3 \| T_2) \oplus F_1, \end{aligned} \quad (22)$$

where $HPW_i^* = h(r_1^* \oplus PW_i)$.

From these relationships, it can be clearly seen that PW_i is protected by an irreversible one-way function, and an attacker cannot extract ID_i, PW_i, r_1^* , and S_g . If an attacker tries to guess the constraint parameters, he must guess all unknown values to verify whether the guessed value is not feasible in polynomial time. If the identity, password, and random number are all N characters and the key of the gateway S_g is M characters, then the probability of guessing the parameters at the same time is about $(1/2^{12N+M})$ [26].

6.1.4. Sensor Device Node Simulated Attack. According to our assumption, an attacker can intercept messages during the execution of the protocol $\{B_1, B_2\}$. After intercepting this message, the attacker attempts to generate another valid message that will be verified by the gateway node GW_j ,

where $B_1 = h(T_4 \| r_5 \| S_{sn} \| ID_{sn} \| T_3 \| SK_i)$, $B_2 = h(r_5 \| T_4) \oplus r_4^*$. However, an attacker cannot calculate effective intercepted messages without knowing the valid SK_i and r_5 , and these messages are protected by the one-way hash function. Therefore, the attacker cannot generate valid other messages. Therefore, our protocol can resist simulated attacks on sensor nodes.

6.1.5. Gateway Node Simulation Attack. In the proposed protocol, the gateway node sends $\{ID_{sn}, R_4, R_5, R_6\}$ and $\{R_4, R_7, R_8\}$ to the sensor and the user. Using these messages, both the sensor node and the user can verify the legitimacy of the gateway node. It is now assumed that an attacker can intercept these two messages.

- (i) Case 1: if the attacker intercepts the message between GW_j and SN_k , namely, $\{ID_{sn}, R_4, R_5, R_6\}$, through the public channel, where $R_4 = h(ID_{sn} \| R_1 \| S_{sn} \| r_4 \| T_3)$, $R_5 = (r_3^* \| T_3 \| r_4) \oplus S_{sn}$, and $R_6 = R_1 \oplus h(ID_{sn} \| h(r_4) \| r_3^*)$, the attacker attempts to generate another message and send it to the sensor node to simulate as a legitimate gateway. However, the calculation of R_4 , R_5 , and R_6 , respectively, depends on the random number r_4 . It should be noted that due to the irreversible properties of the one-way hash function, an attacker cannot extract this value. Since S_{sn} is a shared key parameter between the gateway node and the sensor device node, an attacker cannot guess it in polynomial time.
- (ii) Case 2: if the attacker intercepts the message between GW_j and U_i through the public channel, that is, $\{R_4, R_7, R_8\}$, where $R_7 = h(SK_i \| R_1 \| r_4 \| T_5 \| R_4)$ and $R_8 = (r_5^* \| r_4 \| T_5) \oplus r_3^*$, then the attacker tries to generate another message and transmit it to the user U_i to impersonate legal gateway. However, the calculation of R_7 and R_8 depends on R_1 and r_4 . Also, it should be noted that the attacker cannot extract the values generated due to the irreversibility of the one-way hash function, and these values cannot be guessed in polynomial time. In addition, the user terminated the connection due to an invalid message. Therefore, if an attacker initiates a gateway simulation attack, it may be captured.

6.1.6. Long-Term Key Security. The authentication protocol uses several keys, such as S_{BC} (private key of BC), S_g (the private key of gateway node), and S_{sn} (the shared key between gateway node and sensor device node). It is worth noting that in the setup stage, $S_g = h(ID_g \| S_{BC})$ and $S_{sn} = h(ID_{sn} \| S_{BC})$. Because the keys are protected by a one-way hash function, attackers cannot retrieve them. Similarly, the key of the gateway node S_g cannot be retrieved. Therefore, in the protocol of this chapter, all keys are highly protected.

6.1.7. Mutual Authentication. In this protocol, all entities will authenticate each other to verify the validity of their

identities before the actual information sharing or retrieval occurs. During the implementation of the protocol, the gateway node first authenticates the user's identity according to the received login message $\{ID_{sn}, F_2, F_3\}$, and then the sensor node uses the message $\{ID_{sn}, R_4, R_5, R_6\}$ received from the gateway device node to verify the identity of the gateway node. Similarly, the gateway uses the message $\{B_1, B_2\}$ to authenticate the sensor node, and the user uses the message $\{R_4, R_7, R_8\}$ to authenticate the gateway node. As a result, all participants involved use their own messages to authenticate with each other.

6.1.8. Perfect Forward Confidentiality. This protocol provides perfect forward secrecy, which means that even if one of the long-term keys is disclosed, the session key will not be disclosed. For example, we suppose that the long-term key of the gateway node is disclosed to the attacker in some way. The attacker then attempts to calculate the session key used in the protocol. Even if the secret key is known, the attacker cannot calculate the random number used in the protocol and will not know the shared secret key between the gateway node and the sensor device node. Because the session key depends on a random number, the attacker cannot calculate it. If we assume that the session key used in the protocol has been destroyed by the attacker, the attacker will try to calculate the previous session key. The attacker was unable to calculate the previous session key because he could not extract any confidential information from the compromised session key $SK_i = h(R_1^* \| r_3^* \| r_4^* \| r_5)$. Therefore, our protocol has perfect forward confidentiality.

6.1.9. Effective Authentication. In order to prolong the service life of sensor devices, we hope to reduce the computation cost of sensor and the number of bits it must transmit. In this paper, we also prove that the computation cost of authentication messages of sensor nodes is very low as shown in Table 2. The bits of transmitted message are also less as shown in Table 3. In addition, the sensor node first checks the legitimacy of the user and gateway node by comparing R_4^* with the received R_4 and then performs further calculation and communication processes, which prevent the attacker from repeatedly sending false messages to harm the sensor device node. If the sensor device node participates in the calculation and communication messages as the response of the false message, it will cause unnecessary battery consumption of the sensor device node. Therefore, the protocol provides effective authentication.

6.1.10. Valid Key Changes. When a user suspects that his password has been leaked, the registered user will change their password. Therefore, the protocol proposed in this paper needs the password change function. Registered users can change their passwords in the agreement. Users do not need any support from the gateway node or the registry during the password change process, which reduces the load on the channel and also can resist the DoS attack. In addition, in order to reduce the computation cost, the system

TABLE 2: Computation cost.

	Shi [13]	Choi [14]	Xue [17]	Kumar [19]	Our protocol
U_i	7 H + 3 ECC	12 H + 3 ECC	10 H	8 H	8 H
GW_i	5 H + 1 ECC	5 H + 1 ECC	14 H	9 H	7 H
SN_k	4 H + 2 ECC	7 H + 2 ECC	6 H	5 H	6 H
Total	16 H + 6 ECC	24 H + 6 ECC	30 H	22 H	21 H
Execution time	0.386 s	0.390 s	0.015 s	0.011 s	0.0105 s

TABLE 3: Communication cost.

	Shi [13]	Choi [14]	Xue [17]	Kumar [19]	Our protocol
Total communication cost (bits)	2656	3040	2144	1792	1664
Communication cost of sensor devices (bits)	2656	3040	1440	800	832
Sensor device cost (%)	100	100	67.16	44.64	50

TABLE 4: Comparison of security requirements.

	Shi [13]	Choi [14]	Xue [17]	Kumar [19]	Our protocol
User anonymity	×	×	×	√	√
Resist smart card theft attack	×	×	×	√	√
Resist session key attack	×	√	√	√	√
Resist sensor energy consumption	×	√	√	×	√
Resist internal attacks	√	√	×	√	√
Resist offline password guessing	√	√	×	√	√
Resist replay attacks	√	√	√	√	√
Resistance to man in the middle attacks	√	√	√	√	√
Resist user counterfeiting attack	√	√	√	√	√
Mutual authentication	√	√	√	√	√
Key agreement between user and sensor	√	√	√	√	√

will verify the correctness of the personal information such as the identity and password before calculating the new value with the new password. Therefore, the password change stage is effective and practical.

6.2. Performance Evaluation

6.2.1. Comparison of Security Performance. We compare the security performance of the proposed scheme with the protocols Shi [13], Choi [14], Xue [17], and Kumar [19]. In Table 4, it can be seen that the protocols of Shi [13] are vulnerable to several attacks, such as smart card theft and session key attack. In addition, the protocol of Shi is easy to expose the anonymity of users. From this table, it is clear that our proposed protocol provides strong security protection against related attacks, including user anonymity, password guessing attack, user sensor simulation attack, internal attack, smart card theft attack, and session key disclosure attack. Our improved protocol can provide more adequate security protection, because it can meet all the security requirements. Our protocol is the only one that can resist all known attacks and provide all required security functions.

6.2.2. Comparison of Computation Cost. In Table 2, H, S, and ECC represent the execution time of hash function, symmetric encryption/decryption, and ECC dot product, respectively. The computation amount of user registration is one-time. Therefore, we do not pay attention to this time.

Due to the resource constrained nature of gateway nodes and sensor nodes, we find that Shi et al. [13] used elliptic curve points to calculate authentication messages, and our protocol mainly uses encryption one-way hash function h , XOR“ \oplus ”, and connection” \parallel ” operations to provide security identity authentication. Because the cost of exclusive or and concatenation is negligible, we only consider the cost of the hash function. In addition, the computational complexity of Li [27] can be roughly expressed as $(ECC > S > h)$. As described in Li [27], we assume that one-way hash function (H), symmetric key encryption/decryption algorithm, and ECC of scalar point of elliptic curve need 0.0005, 0.0087, and 0.063075 seconds respectively. From Table 2, we find that the computing cost of our protocol is $(8H + 7H + 6H) = 21 \times 0.0005 = 0.0105$ s, while the computing cost of sensor device node is $6H = 6 \times 0.0005 = 0.003$ s. That is, in our protocol, the computing cost of sensor node is 28% of the total computing cost. Table 2 shows that the computation cost of our protocol is lower than the protocols Shi [13], Choi [14], Xue [17], and Kumar [19]. Therefore, our protocol is suitable for the security authentication of sensor nodes in the medical cyber-physical systems, which can save resources and increase service life.

6.2.3. Comparison of Communication Cost. As shown in Table 3, we compare the communication overhead in this paper with the methods discussed in Shi [13], Choi [14], Xue [17], and Kumar [19]. Communication overhead is the total

number of bits required for transmission during the login and authentication stages. Now, we assume that the participants' identities, random numbers, and timestamps are 32 bits, the results of AES are 512 bits, the ECC points are 320 bits, and the message digests of SHA1 are 160 bits. It can be seen from the results that the total communication cost of our scheme is the lowest, and the cost of sensor devices is low, which can keep the sensor devices active for a long time.

7. Conclusions and Future Works

In this paper, a security authentication model of medical cyber-physical systems based on blockchain is proposed, and the process of data collection and transmission is described in detail. Then, we propose an authentication scheme of sensor devices. The process includes system initialization, user registration, user login, security authentication and key negotiation, password change, and adding sensor nodes. Finally, we analyze the availability and security of the proposed scheme.

Compared with the traditional device identity authentication scheme, this scheme has the following advantages: first, taking the blockchain node as the authentication third party can solve the untrustworthy problem of the third party and also can resist the attacker's attack on the third party's data center to prevent data leakage. Second, the authentication scheme can be adapted to device nodes with different computing, transmission, and storage capacities. At the same time, it can also save the energy consumption of device nodes and increase the service life. Third, the device nodes can be added dynamically. Because the transaction speed of alliance chain is fast, each node has its own private key, the transaction cost is not high, and it cannot be tampered with. However, due to the multiple data types and high complexity of data transactions in the device nodes of the medical cyber-physical systems, and with the needs of the use process, the device nodes need to be added to collect new data. The medical institutions can be connected with the alliance chain, which can provide an innovative way for the medical cyber-physical systems architecture and make the system efficient, safe, and traceable.

Although our scheme has made some progress in the research of device identity authentication, there are still some shortcomings of our work. The following problems need further research:

- (1) A new security authentication protocol for sensor devices in the medical cyber-physical systems is proposed in this paper, which is used to authenticate legitimate users and sensor devices. The protocol realizes the security requirements of authentication process at a lower cost and saves the cost of devices life. Mutual authentication and key establishment can also be completed. In the future, we hope that the scheme of device security authentication can be extended to other application fields to complete the device security authentication with the blockchain technology.
- (2) The security authentication scheme proposed in this paper is based on the blockchain technology.

However, we only analyze the security and effectiveness of the scheme in theory and realizes the simple construction of medical alliance chain. In further, we can use the Hyperledger Fabric to complete more rigorous experimental simulation [28]. The open platform of Hyperledger Fabric, which is open-source and free of charge, provides a modular and scalable architecture and can be used in various industries from banking and health care to supply chain.

Data Availability

No data were used to support the findings of this study.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

The authors would like to thank the colleagues and students in Anhui Provincial Key Laboratory of Network and Information Security. The authors thank the National Natural Science Foundation of China under grant no. 61972438 and Key Research and Development Projects in Anhui Province under grant no. 202004a05020002 for supporting this research.

References

- [1] F. Junior, D. Schneider, and R. Adler, "Dynamic risk management for cooperative autonomous medical cyber-physical systems" in *Proceedings of the International Conference on Computer Safety, Reliability, and Security*, pp. 216–231, Turku, Finland, September 2019.
- [2] D. He and S. Zeadally, "An analysis of RFID authentication schemes for internet of things in healthcare environment using elliptic curve cryptography," *IEEE Internet of Things Journal*, vol. 2, no. 1, pp. 72–83, Feb. 2015.
- [3] Z. Xu, C. Xu, W. Liang, J. Xu, and H. Chen, "A lightweight mutual authentication and key agreement scheme for medical internet of things," *IEEE Access*, vol. 7, pp. 53922–53931, 2019.
- [4] A. Alhayajneh, A. Baccarini, G. Weiss et al., "Biometric authentication and verification for medical cyber physical systems," *Electronics*, vol. 7, no. 12, 436 pages, 2018.
- [5] S. R. Moosavi, T. N. Gia, E. Nigussie et al., "End-to-end security scheme for mobility enabled healthcare Internet of Things," *Future Generation Computer Systems*, vol. 64, pp. 108–124, 2016.
- [6] R. Amin, R. S. Sherratt, D. Giri, S. H. Islam, and M. K. Khan, "A software agent enabled biometric security algorithm for secure file access in consumer storage devices," *IEEE Transactions on Consumer Electronics*, vol. 63, no. 1, pp. 53–61, 2017.
- [7] S. F. Aghili, H. Mala, M. Shojafar, and P. Peris-Lopez, "LACO: lightweight three-factor Authentication, access control and ownership transfer scheme for E-health systems in IoT," *Future Generation Computer Systems*, vol. 96, pp. 410–424, 2019.
- [8] Z. Ma, J. Meng, J. Wang et al., "Blockchain-based decentralized authentication modeling scheme in edge and IoT

- environment”” *IEEE Internet of Things Journal*, vol. 8, no. 4, pp. 2116–2123, 2021.
- [9] Y. Zhang, Y. Qian, D. Wu, M. S. Hossain, A. Ghoneim, and M. Chen, “Emotion-aware multimedia systems security,” *IEEE Transactions on Multimedia*, vol. 21, no. 3, pp. 617–624, 2019.
- [10] Y. Zhang, R. Gravina, H. Lu, M. Villari, and G. Fortino, “PEA: parallel electrocardiogram-based authentication for smart healthcare systems,” *Journal of Network and Computer Applications*, vol. 117, pp. 10–16, 2018.
- [11] S. H. Islam and G. P. Biswas, “Design of two-party authenticated key agreement protocol based on ECC and self-certified public keys,” *Wireless Personal Communications*, vol. 82, no. 4, pp. 2727–2750, 2015.
- [12] S. Chandrasekhar, A. Ibrahim, and M. Singhal, “A novel access control protocol using proxy signatures for cloud-based health information exchange,” *Computers & Security*, vol. 67, pp. 73–88, 2017.
- [13] W. Shi and P. Gong, “A new user authentication protocol for wireless sensor networks using elliptic curves cryptography”” *International Journal of Distributed Sensor Networks*, Article ID 730831, 2013.
- [14] Y. Choi, D. Lee, J. Kim, J. Jung, J. Nam, and D. Won, “Security enhanced user authentication protocol for wireless sensor networks using elliptic curves cryptography,” *Sensors*, vol. 14, no. 6, pp. 10081–10106, 2014.
- [15] F. Chen, Y. Luo, J. Zhang et al., “An infrastructure framework for privacy protection of community medical internet of things,” *World Wide Web*, vol. 21, no. 1, pp. 33–57, 2018.
- [16] H. Shu, P. Qi, Y. Huang et al., “An efficient certificateless aggregate signature scheme for blockchain-based medical cyber physical systems”” *Sensors*, vol. 20, no. 5, 1521 pages, 2020.
- [17] K. Xue, C. Ma, P. Hong, and R. Ding, “A temporal-credential-based mutual authentication and key agreement scheme for wireless sensor networks,” *Journal of Network and Computer Applications*, vol. 36, no. 1, pp. 316–323, 2013.
- [18] Z. Liu, C. Guo, and B. Wang, “A physically secure, lightweight three-factor and anonymous user authentication protocol for IoT,” *IEEE Access*, vol. 8, pp. 195914–195928, 2020.
- [19] D. Kumar, S. Chand, and B. Kumar, “Cryptanalysis and improvement of an authentication protocol for wireless sensor networks applications like safety monitoring in coal mines,” *Journal of Ambient Intelligence and Humanized Computing*, vol. 10, no. 2, pp. 641–660, 2019.
- [20] R. Amin and G. Biswas, “A novel user authentication and key agreement protocol for accessing multi-medical server usable in TMIS”” *Journal of Medical Systems*, vol. 39, no. 3, 33 pages, 2015.
- [21] K. Christidis and M. Devetsikiotis, “Blockchains and smart contracts for the internet of things,” *IEEE Access*, vol. 4, pp. 2292–2303, 2016.
- [22] S. Son, J. Lee, M. Kim, S. Yu, A. K. Das, and Y. Park, “Design of secure authentication protocol for cloud-assisted telecare medical information system using blockchain,” *IEEE Access*, vol. 8, pp. 192177–192191, 2020.
- [23] M. Burrows, M. Abadi, and R. Needham, “A logic of authentication,” *ACM Transactions on Computer Systems*, vol. 8, no. 1, pp. 18–36, 1990.
- [24] S. Kumari and H. Om, “Authentication protocol for wireless sensor networks applications like safety monitoring in coal mines,” *Computer Networks*, vol. 104, pp. 137–154, 2016.
- [25] P. Kumar and H. Lee, “Cryptanalysis on two user authentication protocols using smart card or wireless sensor networks,” in *Proceedings of the IEEE Wireless Advanced (WIAAd)*, pp. 241–245, London, UK, 2011.
- [26] B. Vaidya, D. Makrakis, and H. Moustafah, “Two-factor mutual authentication with key agreement in wireless sensor networks,” *Security and Communication Networks*, vol. 9, no. 2, pp. 171–183, 2016.
- [27] W. Li, Q. Wen, Q. Su, and Z. Jin, “An efficient and secure mobile payment protocol for restricted connectivity scenarios in vehicular ad hoc network,” *Computer Communications*, vol. 35, no. 2, pp. 188–195, 2012.
- [28] X. Cheng, F. Chen, D. Xie et al., “Design of a secure medical data sharing scheme based on blockchain”” *Journal of Medical System*, vol. 44, no. 2, 52 pages, 2020.

Research Article

ECLB: Edge-Computing-Based Lightweight Blockchain Framework for Mobile Systems

Qingqing Xie ¹, Fan Dong ¹, and Xia Feng ²

¹School of Computer Science and Communication Engineering, Jiangsu University, Zhenjiang 212013, China

²School of Automotive and Traffic Engineering, Jiangsu University, Zhenjiang 212013, China

Correspondence should be addressed to Qingqing Xie; xieqq@ujs.edu.cn

Received 19 February 2021; Revised 6 April 2021; Accepted 16 April 2021; Published 28 April 2021

Academic Editor: Lu Liu

Copyright © 2021 Qingqing Xie et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The blockchain technology achieves security by sacrificing prohibitive storage and computation resources. However, in mobile systems, the mobile devices usually offer weak computation and storage resources. It prohibits the wide application of the blockchain technology. Edge computing appears with strong resources and inherent decentralization, which can provide a natural solution to overcoming the resource-insufficiency problem. However, applying edge computing directly can only relieve some storage and computation pressure. There are some other open problems, such as improving confirmation latency, throughput, and regulation. To this end, we propose an edge-computing-based lightweight blockchain framework (ECLB) for mobile systems. This paper introduces a novel set of ledger structures and designs a transaction consensus protocol to achieve superior performance. Moreover, considering the permissioned blockchain setting, we specifically utilize some cryptographic methods to design a pluggable transaction regulation module. Finally, our security analysis and performance evaluation show that ECLB can retain the security of Bitcoin-like blockchain and better performance of ledger storage cost in mobile devices, block mining computation cost, throughput, transaction confirmation latency, and transaction regulation cost.

1. Introduction

Since Satoshi Nakamoto invented Bitcoin in 2008 [1], the blockchain technology has gained considerable interest and adoption in multiple fields, such as economics, cryptography, and mathematics. Blockchain makes it possible to process the online trade among mutual distrust parties. The security of the blockchain technology is achieved by sacrificing prohibitive computation and storage resources to jointly maintain a unique transaction ledger. However, most mobile systems are underresourced due to weak mobile devices. As a result, it is a matter of great difficulty to apply the blockchain technology to mobile systems. Some details are shown as follows:

On the one hand, each miner contributes immense computation effort to painstakingly solve a cryptographic problem, i.e., the proof of work (PoW) problem. Only the

miner who first succeeds in solving the PoW problem can pack some transactions into a new valid block and append to the longest ledger. Generally, the mining machines, such as ANTMINER S9 Hydro, reach up to 18TH/s [2], while the hash rates of the normal mobile devices are just at the MH/s level. According to the statistics, Bitcoin alone is estimated to use tens of Terawatt hours per year, which is enough to power a mid-sized country. It indicates that most normal mobile devices cannot undertake the mining work because of their limited computational power.

On the other hand, each miner has to maintain an entire copy of the transaction ledger, i.e., every transaction record from the beginning of time. Storing the entire blockchain ledger requires a remarkable amount of storage capacity. Take Bitcoin as an example; the total size of a local ledger reached more than 380 GB on February 18, 2021 [3]. And, it is growing at a rate of around 70 MB per day. It is not feasible

for a normal mobile device to store such a large-size ledger. Table 1 shows some ledger growth information in several different blockchain systems.

In conclusion, most mobile devices are unable to provide such computation and storage capacities to meet the requirements for working as miners. The aforementioned issues must be solved to popularize the blockchain applications in mobile systems.

Edge computing appears with inherent decentralization and strong resources, which can provide a natural solution to overcoming the aforementioned resource-insufficiency situation [7, 8]. As Abbas et al. pointed out in [9], edge computing is now a promising technology in the 5G mobile environment. Each edge node locates close to the end devices at the edge network and can provide sufficient capacities of storage, computation, and networking to support the mining work. That is, edge computing relies on edge nodes to create services that are distributed across edge domains. Thus, constructing a lightweight blockchain system based on edge computing is a natural and appropriate way to make the blockchain technology widely used in practical mobile systems.

1.1. Challenge. The direct integration of blockchain and edge computing can only relieve some storage and computing pressure at end mobile devices. But first, the end devices do not always work as completely light nodes. They usually are interested in some types of transaction information, maybe related to their jobs, life, or something else. They often want to store some transactions as well. Second, there are some other open problems, such as improving the transaction confirmation latency, throughput, etc. These performance metrics must be optimized when applying the blockchain technology to mobile systems. It is a paradox. The reason is that, on the one hand, the high transaction confirmation latency and low throughput are caused by the computation-intensive consensus protocol itself. On the other hand, the computation-intensive consensus protocol is a key to maintaining the security and stability of blockchain systems.

To sum up, the challenge is how to solve this paradox to achieve both light weight at end mobile devices and superior performance regarding transaction confirmation latency and throughput.

Some related works have been done so far. Cebe et al. [10] proposed an integrated lightweight blockchain framework for forensic applications of connected vehicles, abbreviated as Block4Forensic. In Block4Forensic, each node maintains a shared ledger and a fragmented ledger. The shared ledger stores hash values. The fragmented ledger stores some transactions attracted by the corresponding participant. Liu et al. [11] proposed a mobile edge-computing-enabled wireless blockchain framework where the computation-intensive mining tasks could be offloaded to the nearby edge nodes and the cryptographic hashes of blocks could be cached in the edge servers. Chen et al. [12] proposed a multi-hop cooperative and distributed computation offloading algorithm that considered the data processing tasks and the mining tasks together for blockchain-empowered Industrial

TABLE 1: Ledger growth information in several different blockchain systems, according to the statistics on February 18, 2021 [3].

Blockchain	Block interval	Block count	Ledger size (GB)
Bitcoin [1]	10 m·45 s	671,200	382.04
Ethereum [4]	13.3 s	11,884,711	608.20
Bitcoin cash [5]	10 m·40 s	675,436	184.14
Litecoin [6]	2 m·28 s	2,003,322	40.64

Internet of Things (IIoT). Lei et al. [13] proposed Groupchain, a novel scalable public blockchain of a two-chain structure suitable for fog computing. To some extent, Groupchain overcomes the scalability challenge of blockchain's integration with fog computing. Eyal et al. [14] proposed Bitcoin-NG, a Byzantine fault-tolerant blockchain protocol. It decouples Bitcoin's blockchain operation into leader election and transaction serialization. It introduces high generation frequency of micro-blocks for transaction commitment. Table 2 shows the advantages and disadvantages of these works. These works give us great inspiration to study the blockchain application problems. There are also some other related works [15–17]. All these works can only solve part of the aforementioned challenges.

1.2. Contributions. The main contributions are summarized as follows:

- (1) We propose a novel lightweight blockchain framework based on edge computing (ECLB) for mobile systems. It takes edge nodes as miners, to relieve some storage and computation pressure at end mobile devices. As for the mobile devices, we introduce the fragmented ledger structure [10], to let them obtain the transaction information of interest. In this proposed ECLB framework, edge computing and blockchain technology complement each other, which makes the blockchain technology applicable in mobile systems.
- (2) Under the ECLB framework, we reform the block structures into leader block and transaction block. The leader blocks are used to record leader nodes, who succeed in solving the PoW puzzles. The transaction blocks are used to record the transaction history via most edge nodes' signature assurance. Such a structure optimizes the blockchain metrics, including throughput and transaction confirmation latency.
- (3) Considering the popular permissioned blockchain settings, we specifically utilize symmetric encryption algorithm and ciphertext-policy attribute-based encryption (CP-ABE) scheme [18] to design a plugable regulation layer. It is a secure solution for supervising the transaction behaviors. Note that due to the low efficiency, the CP-ABE schemes cannot be readily adopted. Here, in order to meet the requirement of high efficiency, we combine CP-ABE with symmetric encryption algorithm to improve the regulation efficiency.

TABLE 2: Advantages and disadvantages of some existing works.

Research	Advantages	Disadvantages
[10]	Make each end device to maintain a fragmented ledger, to reduce the storage pressure	
[11]	Offload the computation-intensive mining tasks to nearby edge-computing nodes	Does not consider improving the transaction throughput and confirmation latency
[12]	Disburden the data processing tasks and mining tasks from end devices to edge servers	
[13]	Employ a leader group to optimize the transaction throughput and confirmation latency	Does not consider reducing the ledger storage pressure at end devices
[14]	Decouple Bitcoin's blockchain operation into leader election and transaction serialization to achieve scalability	

- (4) We analyze the security to demonstrate that our ECLB achieves fault tolerance, high security level with 16 edge nodes, Sybil attack resistance, double-spending attacks resistance, and chosen-plaintext attack (CPA) resistance. We also conduct performance evaluation, demonstrating that ECLB achieves lower cost of ledger storage and block mining computation, and better throughput, transaction confirmation latency, and regulation efficiency.

1.3. Structure of the Paper. The rest of the paper is organized as follows: Section 2 presents the preliminaries. Section 3 presents our ECLB system model. Section 4 presents our ECLB protocol design. Sections 5 and 6 formally analyze the security and experimentally evaluate the performance of our ECLB. Section 7 reviews the related works. Section 8 discusses the solution to the blockchain fork problems. Finally, Section 9 concludes this paper.

2. Preliminaries

We briefly review the blockchain technology and the CP-ABE scheme.

2.1. Blockchain. Blockchain was invented by Satoshi Nakamoto in 2008 to serve as the public transaction ledger of the Bitcoin cryptocurrency [1]. The ledger records a continuously growing list of transactions, called blocks, which are linked by the cryptographic hash of the previous block. The general structure of the block and the blockchain is shown in Figure 1.

A blockchain is typically managed by a peer-to-peer (P2P) network collectively following a predefined consensus protocol. Each miner contributes a large amount of computation energy for packing transactions into a new block, i.e., the consensus procedure or mining tasks. As we know, PoW is a frequently and widely used consensus protocol, such as in the Bitcoin systems. PoW requires a complicated computational process for packing transactions. It is a random process where a lot of trials and errors are required on average before a PoW solution is generated. In PoW, all the miners have to use different nonces and calculate the hash value of the constantly changing block header continuously, until the calculated hash value is not greater than a

given value. When one node obtains the target, all other nodes must mutually confirm the correctness of the value. Finally, a new block is generated. The flow of new block generation procedures is shown in Figure 2. A new block is determined in a round.

The characteristics of the blockchain technology are listed as follows:

- (i) Decentralization: the blockchain is built on a P2P network, which is naturally decentralized. All participating nodes have the same copy of the blockchain ledger.
- (ii) Immutability: once a block is written to a blockchain, the information cannot be altered.
- (iii) Authenticity: users can trust that transactions will be executed exactly as the protocol comments. Thus, the transaction data in blockchain ledger are all authentic.
- (iv) Pseudonymity: blockchain uses a pseudo-identity mechanism. Each user can generate as many pseudo-identities as he/she likes to increase identity privacy.

Obviously, it should reduce the pressure of both ledger storage and block mining computation to design a thoughtful lightweight blockchain system. Simultaneously, the scalability is also an important factor to measure a blockchain system. Scalability itself includes two important metrics: throughput and transaction confirmation latency.

2.2. CP-ABE. The CP-ABE scheme was proposed to achieve fine-grained access control [18]. In CP-ABE, a user's secret attribute key is associated with an attribute set. The ciphertext of a message is associated with an access policy. A user will succeed in decrypting a ciphertext if and only if the user's attribute set matches the access policy associated with the ciphertext.

The CP-ABE scheme consists of four algorithms [18]:

- (i) $\text{CPABE.Setup}(1^\lambda) \rightarrow \text{MK, PK}$: it takes the security parameter 1^λ as input, and outputs a master key MK and a public key PK.
- (ii) $\text{CPABE.Encrypt}(T, m, \text{PK}) \rightarrow \text{CT}$: it takes as input an access policy tree T over the universe of attributes, a message m , and the public key PK, then encrypts m

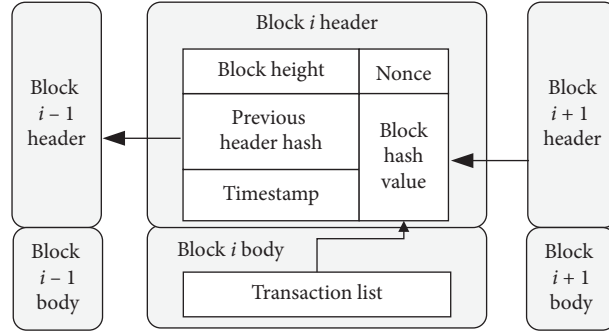


FIGURE 1: The chain structure of the blockchain.

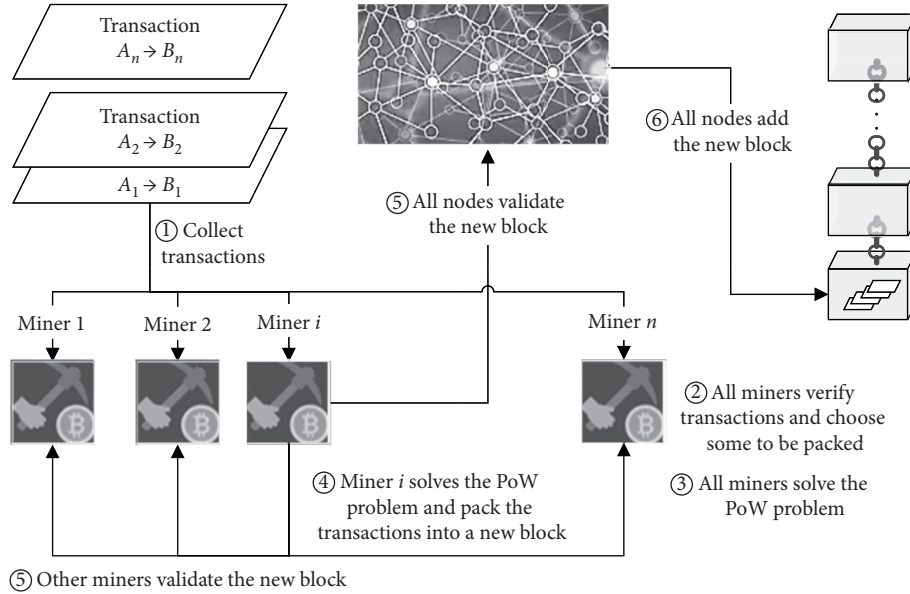


FIGURE 2: The flowchart of new block generation procedures.

as a ciphertext CT , such that only a user that possesses a set of attributes that satisfies the access tree T will be able to decrypt the message m .

(iii) $CPABE.KeyGenerate(PK, MK, A_u) \rightarrow SK_u$: it takes as input the public key PK , the master key MK , and a user's attribute set A_u , then outputs the user's secret attribute key SK_u .

(iv) $CPABE.Decrypt(PK, CT, SK_u) \rightarrow m$: it takes as input the public key PK , the ciphertext CT of a message m , and a user's secret attribute key SK_u , then outputs the message m if the user's attribute set satisfies the access policy tree associated with CT .

3. System Model

The conception model of our ECLB framework is shown in Figure 3. It mainly consists of the following four layers:

(1) Cloud data center layer: it is in charge of storing encrypted transaction information specifically for the permissioned blockchain setting. We assume that the cloud data center is honest but curious. That

means, it acts in an honest fashion and correctly follows the designated protocol specification. However, it is curious to infer and analyze the stored data to harvest additional information to gain illegal profits.

- (2) Edge nodes layer: each node on this layer undertakes the mining work as a blockchain miner node, i.e., solving the PoW puzzles and storing an entire copy of the blockchain ledger. Each edge node i has a public/private key-pair (pk_i, sk_i) . It is either honest or Byzantine. Byzantine nodes do not follow the consensus protocol accidentally or maliciously. It means that they might fail to join the consensus or collude to attack the whole network. Assume that there are n edge nodes, these n nodes are well connected in a P2P network, and the number of Byzantine nodes is f , where $n \geq 3f + 1$ is required in our model [13].
- (3) End devices layer: it consists of some traditional PC or mobile computing end devices, such as laptop, smart phone, etc. They usually provide weak capacities of computing, storage, and networking.

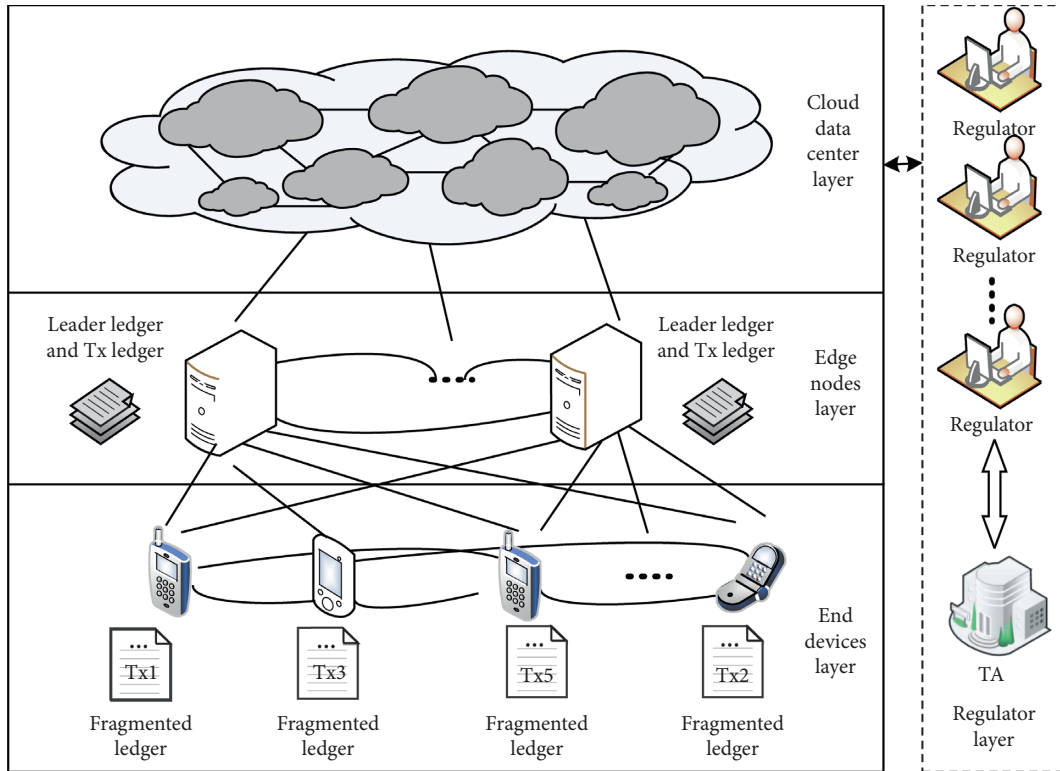


FIGURE 3: Our ECLB model.

Hence each end device only stores a fragmented ledger [10], consisting of the copy of the block headers and some transaction records of interest. End devices are usually too weak to be miners. They only download some transaction information of interest from nearby edge nodes. Thus, they can be trustworthy or not, which has no effect to the whole network.

- (4) Regulator layer: it consists of some regulators and a trusted authority (TA). This layer is designed specifically for the permissioned blockchain setting. On the one hand, the regulators request to gain the transaction data in cloud to carry out trading regulation. On the other hand, considering the transaction privacy preservation, only the regulators are allowed to get the transaction data. And, they are not allowed to get the data outside of their privileges. Thus, the regulators are assumed to be honest but curious. The TA is in charge of controlling the access privilege, i.e., authorizing the access privilege only to the regulators. The TA is assumed to be trustworthy.

4. ECLB Protocol Design

In this section, we will describe our ECLB protocol design in detail, including three parts: transaction ledger storage, transaction packing and confirmation, and transaction regulation. Some major notations used in our ECLB protocol are shown in Table 3.

TABLE 3: Some major notations used in our ECLB protocol.

Notations	Description
Node _{<i>i</i>}	The <i>i</i> -th edge node
(<i>pk_i</i> , <i>sk_i</i>)	The public and private key pair of Node _{<i>i</i>}
<i>n</i>	The number of edge nodes
<i>b_t</i>	The candidate transaction block generated at time <i>t</i>
Sign(<i>·</i> , <i>·</i>)	A signing algorithm
Verify(<i>·</i> , <i>·</i>)	A signature verification algorithm corresponding to Sign(<i>·</i> , <i>·</i>)
<i>s_t^{<i>i</i>}</i>	Node _{<i>i</i>} 's signature on <i>b_t</i> , i.e., <i>s_t^{<i>i</i>}</i> = Sign(<i>sk_i</i> , <i>b_t</i>)
UTXO	The unspent transaction output (UTXO) set
<i>vr_t</i>	The verification result of the candidate block <i>b_t</i>
APT	An access policy tree
PK	A public key in the regulator layer
MK	A master key of the trusted authority in the regulator layer
key	A symmetric key
CT _{key}	The ciphertext of key
SE(<i>·</i> , <i>·</i>)	A symmetric encryption algorithm
SD(<i>·</i> , <i>·</i>)	A symmetric decryption algorithm
<i>tx_i</i>	The <i>i</i> -th transaction record
CTX _{<i>i</i>}	The ciphertext of the transaction record <i>tx_i</i>

4.1. Transaction Ledger Storage. In real applications, the edge nodes are located close to the end mobile devices, and have much stronger storage and computation capabilities compared with the end mobile devices. Thus, we take the edge nodes as blockchain miners and the edge devices as light nodes.

Specifically, in our framework, there are two chains: a leader chain and a transaction (Tx) chain. There are two kinds of blockchain ledgers: full ledger and fragmented

ledger. The full ledger records the identities of both the leaders and the transaction history, by packing the public keys of the leaders and the transaction records. The fragmented ledger records the block headers of the full ledger and some transaction records attracted to the corresponding end mobile devices. Obviously, the fragmented ledger [10] is specifically introduced for the end devices. Each edge node stores an entire copy of the full ledger. The structure of the transaction ledger storage is described in Table 4.

As their name imply, the *leader chain* packs the public keys of the leader, while the *Tx chain* packs the whole transaction records. The ledgers produced by both the leader chain and the Tx chain form the *full ledger*. All the block headers in the full ledger and a part of the transactions packed by the Tx chain form the *fragmented ledger*. Obviously, the size of the fragmented ledger is much smaller than that of the full ledger. The *edge nodes* play the role of *miner nodes*, and thus are responsible to store the full ledger. The *end mobile devices* only need to store the fragmented ledger, due to their weak resources. Thus, they play the role of *light nodes*. Nevertheless, they still can obtain the transaction information since the fragmented ledger maintains a part of transactions.

4.2. Transaction Packing and Confirmation. Section 4.1 introduces lightweight ledger storage at end mobile devices. In this part, we will describe the scalability optimization and lightweight mining computation.

Inspired by [13, 14], we construct a leader group to achieve high scalability. The edge nodes participating in the mining work form the leader group. Assume that there are n edge nodes (i.e., miners) that collectively commit transactions via new blocks, and at most $(n-1)/3$ of them are Byzantine nodes.

In our ECLB, there are two chains growing in parallel: a leader chain and a transaction (Tx) chain, as shown in Figure 4. For convenience, we simply call the blocks in the leader chain *leader blocks*, and the blocks in the Tx chain *Tx blocks*. The leader chain is used to record which edge node competes successfully for serving as a leader.

In our ECLB, first each edge node tries to solve a PoW problem to mine a leader block for competing for being a leader. The leader block packs its own public keys and the corresponding reward coinbase. Once an edge node wins, denoted as Node_i , a new leader block will be generated and broadcast to all the other edge nodes. Node_i chooses and packs some new transactions into a Tx block by embedding its signature as assurance. In parallel, all the edge nodes still can work on solving another PoW problem to compete for being a leader. Once another edge node wins, denoted as Node_j , Node_j will be a new leader and the aforementioned procedures are repeated. Note that an edge node can be a leader in succession, i.e., $i = j$ may happen. We can see that only the leader has the right to pack new transactions.

Now we present the aforementioned transaction packing and confirmation process, as follows:

- (1) To compete for being a leader, each edge node works on mining a leader block by solving a PoW problem. Once an edge node succeeds in solving the PoW and gets a valid leader block, it immediately broadcasts the leader block to all the other edge nodes. Assume that Node_i is the winner. All the other nodes check its validity and append the leader block to the local leader chain if it is valid. In parallel, all the edge nodes still can work on mining a new leader block based on the latest leader chain, for replacing the original leader.
- (2) The leader, i.e., the winning edge node Node_i , first packs a set of new transactions, then computes a signature s_i^j , and finally generates a corresponding new candidate Tx block b_t and broadcasts b_t to the other edge nodes. The candidate Tx block generation algorithm is shown in Algorithm 1.
- (3) Once receiving a candidate Tx block b_t generated by the leader, each other edge node Node_j verifies b_t based on the signatures and UTXOs, where $j = 1, 2, \dots, i-1, i+1, \dots, n$. The verification algorithm of the candidate Tx block is shown in Algorithm 2. If Node_j verifies that the candidate Tx block is valid, it will sign the candidate block b_t as $s_t^j = \text{Sign}(sk_j, \text{hash}(b_t))$, and broadcast s_t^j to other nodes.
- (4) All the edge nodes collect the signed block from each other edge node. If an edge node obtains the signed block b_t from 2/3 supermajority, meaning that all the edge nodes agree on the candidate block b_t , then b_t will be appended to the Tx chain.
- (5) Repeating steps (2)–(4) until another leader block is generated. That is, during the steps (2)–(4), in parallel, all the edge nodes work on solving a PoW problem and mining a new leader block to compete for being a leader.

The transaction packing and confirmation processes are shown in Figure 5, assuming that the edge node Node_0 is the leader who is the first to succeed in solving PoW, Node_3 is faulty.

4.3. Transaction Regulation. In public/permissionless blockchain systems, any transaction information is available to any entity in the network, which provides much convenience to the regulator department. However, in the permissioned blockchain, only the blockchain member nodes are allowed to obtain the transaction information. Hence, an interface of reading the Tx ledger needs to be set for outside regulator department. To this end, we will design a transaction regulation protocol specifically for the permissioned blockchain setting.

Considering the requirements of both privacy preservation and secure regulation, we will employ the CP-ABE scheme to realize secure sharing of the transaction records with legal regulators. However, the CP-ABE scheme is

TABLE 4: Transaction ledger storage at edge nodes and end devices.

Roles	Node types	Ledger types	Ledger contents
Miner nodes	Edge nodes	Full ledger	The public keys of the leaders and the whole transaction records
Light nodes	End mobile devices	Fragmented ledger	The block headers and some transactions of interest

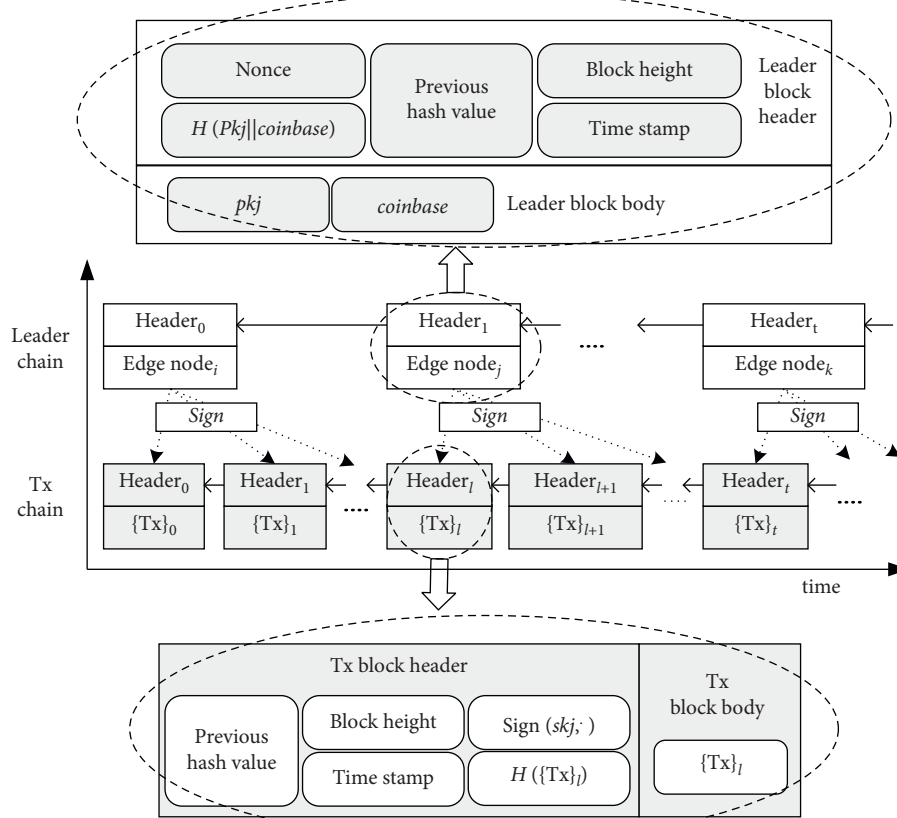


FIGURE 4: Two-chain structure of ECLB.

Input:

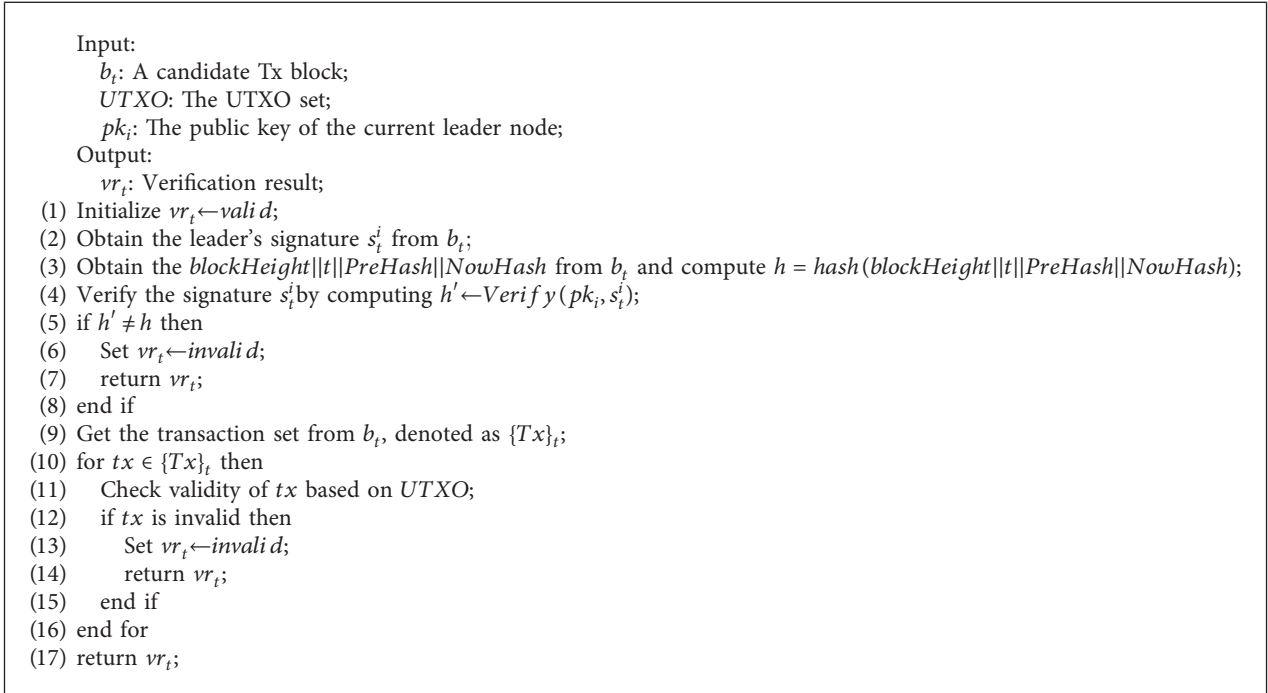
- $\{Tx\}_{new}$: The set of new transactions;
- PreHash: Hash of previous block header;
- t : Timestamp;
- sk_i : Secret key of the leader;

Output:

b_i : Candidate block;

- (1) Select a set of valid transactions from $\{Tx\}_{new}$, denoted as $\{Tx\}_i$;
- (2) Set $body \leftarrow \{Tx\}_i$;
- (3) Construct a Merkle hash tree MT over $\{Tx\}_i$, and denote its root hash as *NowHash*;
- (4) Increase $blockHeight \leftarrow blockHeight + 1$, where $blockHeight$ represents the block height and sets zero in the genesis block;
- (5) Compute a signature $s_i^j = \text{Sign}(sk_i, \text{hash}(blockHeight || t || \text{PreHash} || \text{NowHash}))$;
- (6) Set $header \leftarrow \{blockHeight, t, \text{PreHash}, \text{NowHash}, s_i^j\}$;
- (7) Set $b_i \leftarrow header, body$;
- (8) return b_i ;

ALGORITHM 1: Candidate Tx block generation.



ALGORITHM 2: Candidate transaction block verification.

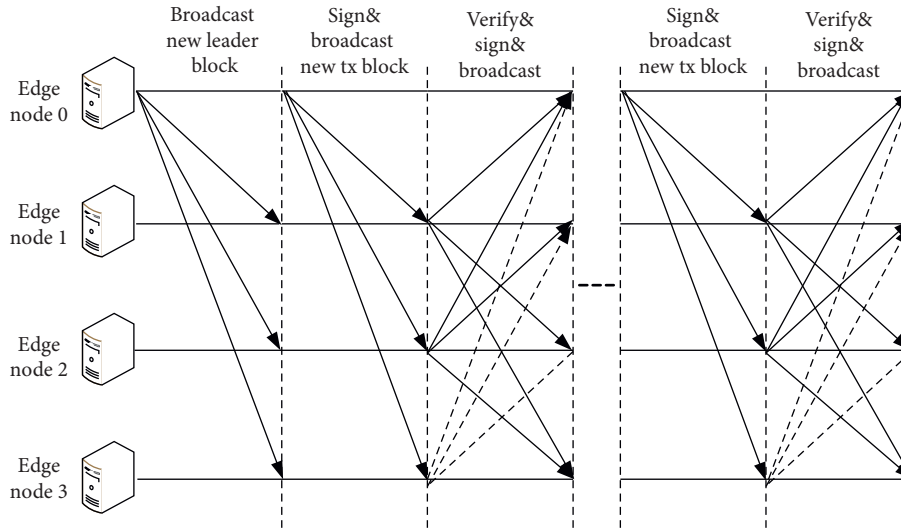


FIGURE 5: The flowchart of the transaction packing and confirmation protocol.

notoriously inefficient in encryption and decryption. To solve this problem, we will utilize the key encapsulation mechanism to improve the efficiency [19, 20]. First, the central control of the permissioned blockchain encrypts a symmetric key key using the CP-ABE scheme. Then, key is shared with all the edge node members and repeatedly used to encrypt the valid and newly packed transaction records. Last, only the designated data consumers, i.e., valid regulators, can succeed in decrypting the key, and further decrypting the transaction records by the key. As a result, the transaction records are stored in ciphertext format in cloud

server and can only be accessed by the legal regulators. The detailed transaction regulation protocol consists of the following steps:

- (1) The central controller of the permissioned blockchain generates a symmetric key key and determines an access policy tree APT . Then, it calls the $CPABE.\text{Encrypt}$ algorithm to encrypt key under APT , as

$$CT_{key} = CPABE.\text{Encrypt}(APT, key, PK) \quad (1)$$

CT_{key} is outsourced to the cloud for storage. In addition, the central controller sends the symmetric key key to all the edge nodes.

- (2) A regulator requests a secret attribute key SK_u from the trusted authority (TA) in the regulator layer. The TA calls the CP-ABE's key generation algorithm to compute

$$SK_u = \text{CPABE.KeyGenerate}(\text{PK}, \text{MK}, A_u) \quad (2)$$

where A_u is the regulator's attribute set. SK_u is sent to the corresponding regulator.

- (3) The regulator downloads the key ciphertext CT_{key} from the cloud, and uses his or her secret attribute key SK_u to decrypt the symmetric key key, i.e.,

$$\text{key} = \text{CPABE.Decrypt}(\text{PK}, CT_{key}, SK_u) \quad (3)$$

If his or her attribute set A_u satisfies the access policy tree APT, he or she will obtain key, otherwise null.

- (4) Once a new Tx block b_t is committed, the corresponding leader uses the symmetric key key to symmetrically encrypt each transaction record tx_i of b_t as

$$CTX_i = \text{SE}(\text{key}, tx_i), \quad (4)$$

where $\text{SE}(\cdot, \cdot)$ represents a symmetric encryption. Each CTX_i is outsourced to the cloud server. This step is repeated with each new committed Tx block.

- (5) The regulator downloads the transaction ciphertext CTX_i from the cloud, and symmetrically decrypts it by key to obtain the plain transaction records, i.e.,

$$tx_i = \text{SD}(\text{key}, CTX_i), \quad (5)$$

where $\text{SD}(\cdot, \cdot)$ is the symmetric decryption algorithm corresponding to $\text{SE}(\cdot, \cdot)$.

Note that the aforementioned steps (1) and (3) are, respectively, one-time computation during the symmetric key's life cycle. It can be set very long until *key* is leaked. Step (2) is also a one-time computation for each regulator. Hence, the online computation cost of this transaction regulation module mainly depends on steps (4) and (5). These two steps are symmetric encryption and decryption, which are efficient obviously. Thus, the real-time transaction regulation is well supported in our ECLB.

In conclusion, we design an efficient transaction regulation module specifically for the permissioned blockchain setting, by combining the CP-ABE scheme with the key encapsulation mechanism. This transaction regulation module preserves the transaction privacy preservation and simultaneously supports efficient regulation required by the practical government department.

5. Security Analysis

In this section, we will provide some security analysis, including fault tolerance, the least number of edge nodes to

reach a high security level, Sybil attack, double-spending attack, and chosen-plaintext attack (CPA).

5.1. Fault Tolerance. The security of fault tolerance is analyzed by proving the following theorem.

Theorem 1. *The edge nodes guarantee fault tolerance, if the number of Byzantine edge nodes f is no more than $(n-1)/3$, i.e., $n \geq 3f + 1$, where n is the total number of edge nodes.*

Proof. Assume all the edge nodes are divided into three disjoint sets, i.e., H_1, H_2, B , where H_1 and H_2 represent two sets of honest edge nodes and B are all Byzantine nodes. Thus, we have

$$|H_1| + |H_2| + |B| = n, \quad (6)$$

and for the worst case,

$$|B| = f. \quad (7)$$

If the Byzantine edge nodes in B want to change the system status, they need to first mine a leader block to propose a consensus process. In this way, malicious nodes can gain agreement from supermajority edge nodes. To win this attack, it requires

$$|H_1| + |B| \geq n - f, \quad (8)$$

$$|H_2| + |B| \geq n - f. \quad (9)$$

By simplifying equations (6)–(9), we can get

$$f \geq n/3. \quad (10)$$

Therefore, all the edge nodes are able to guarantee fault tolerance if the number of Byzantine members f is no more than $(n-1)/3$, i.e., $n \geq 3f + 1$. \square

5.2. The Number of Edge Nodes. We assume that each edge node is either honest or Byzantine, and the mining is a fair game. Let p be the probability of that an edge node is Byzantine. As mentioned in Section 5.1, there are less than $f = (n-1)/3$ Byzantine edge nodes. Thus, using the cumulative binomial distribution, the security probability of the leader chain is computed as

$$P[X \leq f] = \sum_{k=0}^f \binom{n}{k} p^k (1-p)^{n-k}. \quad (11)$$

Considering that in the Bitcoin, the recommended 6-block-confirmation is calculated under $p = 0.1$ and security level ≥ 0.99 , we will set the same adversary probability and security level. The leader chain ensures the same security as long as there are not less than 16 edge nodes, as shown in Figure 6. It means that as long as our ECLB framework is configured with no less than 16 edge nodes, the security level ≥ 0.99 can be guaranteed.

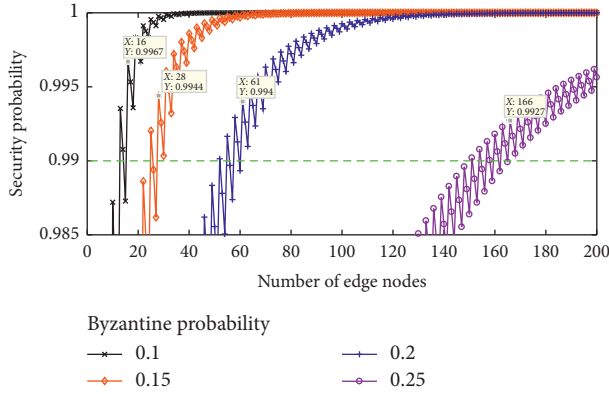


FIGURE 6: Security under different byzantine probabilities.

5.3. Sybil Attack. Sybil attacks [21] allow a malicious participant to subvert a peer-to-peer network by creating many pseudonymous identities in order to work as multiple distinct nodes.

By using PoW to compete for being a leader, the leader chain has a natural ability to resist Sybil attacks. Recall that once an edge node becomes a leader, it is the only one to be allowed to broadcast blocks. In order to become a leader, it must solve a PoW problem, which is extremely computationally intensive. PoW raises the cost of creating a new leader identity. Thus, it mitigates Sybil attacks, wherein security property is guaranteed by the leader chain.

5.4. Double-Spending Attacks. In the leader chain, any edge node checks the collective signatures of a Tx block, in which a supermajority (i.e. $1/2$) of the edge nodes permit its validity. In other words, the Tx chain is under the supervision of all the edge nodes instead of a single leader. Thus, a double-spending attacker will have no chance to use the same coin(s) to issue two (or more) transactions [22]. Moreover, in this respect, 0-block-confirmation services can be provided for clients in a secure way.

5.5. Chosen-Plaintext Attack (CPA). We first give Definition 1 of CPA security of our ECLB protocol. We then demonstrate the CPA security of our ECLB protocol by proving Theorem 2.

Definition 1. Our ECLB protocol is CPA secure if the transaction regulation protocol is CPA secure.

Theorem 1. Our ECLB protocol is CPA secure.

Proof. We reduce the CPA security proof of our ECLB protocol to that of the transaction regulation protocol. As we know, there are some efficient and symmetric encryption algorithms that are secure against CPA, such as AES and DES. Hence, whether the transaction regulation protocol is secure against CPA depends on the indistinguishability of the symmetric key's ciphertext against CPA. The indistinguishability of the symmetric key key's ciphertext is

guaranteed by the CPA security of the traditional CP-ABE scheme [18]. Thus, it proves that the transaction regulation protocol is secure against CPA. Finally, according to Definition 1, our ECLB protocol is also CPA secure. \square

6. Performance Evaluation

6.1. Implementation. We extend the Bitcoin Simulator [23] to implement the key elements of the transaction packing and confirmation process for performance analysis, with the absence of Byzantine nodes. The transaction regulation protocol is implemented using the Java Pairing-based Cryptography (JPBC) library [24]. The experimental machine is configured with Intel(R) CORE(TM)2 Duo CPU E8400 @ 3.00 GHz and 8.00 G RAM. In addition, we simulate the broadcast, sign, and verify procedures by imposing a latency of 100 ms for each edge node [13]. The reason is that the network topology is almost a complete graph, and the broadcast procedure is very fast.

6.2. Ledger Storage. We set the size per transaction at around 256 bytes, and the size per block at 1 MB. Thus, one block contains around 4000 transactions. In Bitcoin, each full node, i.e., miner, stores the entire transaction ledger, while each light node stores only the block headers. In ECLB, each edge node stores the entire leader ledger and transaction ledger, while each end mobile device stores the fragmented ledger, i.e., only all the block headers and some transaction of interest. Note that the leader ledger is very small compared with the transaction ledger, since only one leader block is mined after around every 1500 transaction blocks. Hence, we speculate that the ledger storage cost at an edge node in ECLB is almost as high as that at a full node in Bitcoin. The ledger storage cost at the end mobile device in ECLB will be slightly higher than that at the light node in Bitcoin but much lower than that at the full node and the edge node.

Figure 7 actually demonstrates the aforementioned speculation, where “ $x\%$ ToI” represents an average $x\%$ percentage of transactions stored by each end mobile device. The ledger storage cost at light node of Bitcoin is too small to be shown. Even though the ledger storage cost at end devices is slightly higher than that at light node in Bitcoin, for 3.6×10^6 transactions, it costs around 86 MB to store the fragmented ledger at the end device with 10% transaction of interest. Thus, our ECLB achieves lightweight ledger storage at end mobile devices.

6.3. Block Mining. In Bitcoin, each miner needs to solve a PoW problem for mining a new block. While in our ECLB, only the leader block is mined through solving a PoW problem. All the Tx blocks are created by only the corresponding signatures, which is much lighter than solving a PoW problem. Most importantly, the leader block mining and the Tx block creation procedures are executed in parallel. Thus, our ECLB holds lightweight and efficient block mining process. Figure 8 shows the block mining time with

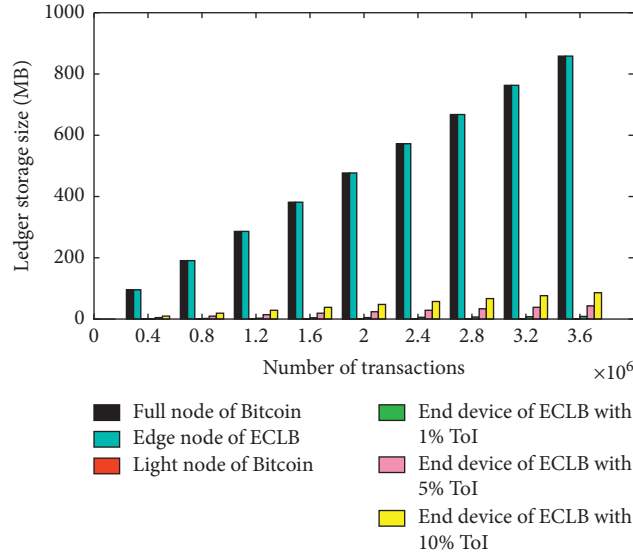


FIGURE 7: Ledger storage size with different number of transactions to pack.

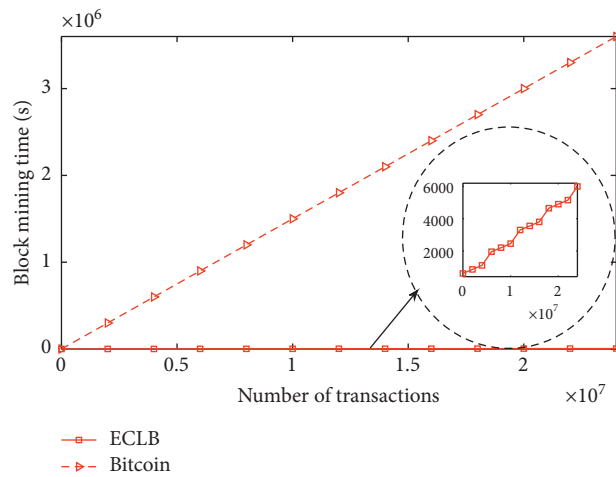


FIGURE 8: Block mining time with different number of transactions.

different number of transactions. It sufficiently demonstrates that our ECLB provides lightweight and efficient block mining.

6.4. Throughput. We set block frequency to 1 per 10 minutes for Bitcoin and the leader block frequency as the same. Obviously, the throughput of our ECLB is shown by only the Tx chain. We test the throughputs with different block sizes. Figure 9 shows the experimental results. We observe that our ECLB achieves much higher throughput than Bitcoin of 100 times on average.

6.5. Transaction Consensus Latency. Since the transaction commitment is submitted through the Tx chain, we only consider the transaction block commitment among the edge nodes for the transaction consensus latency. To see the

scalability of ECLB’s consensus process in terms of the number of edge nodes, we set the transaction block size to 1 MB, which is the maximum block size in current Bitcoin. In Bitcoin, the consensus latency is the time for at least 50% nodes to receive a block. Groupchain [13] and our ECLB have 3 and 2 rounds of interactions on average, respectively. Figure 10 shows the experimental results. We observe that our transaction consensus latency is slightly higher than that of Bitcoin and lower than that of Groupchain. But Groupchain and our ECLB allow the blocks already appended to the blockchain to be confirmed valid immediately without the 6-block confirmation, while the Bitcoin needs 6-block-confirmation mechanism.

6.6. Regulation Efficiency. We evaluate the regulation efficiency from the aspects of online transaction encryption and decryption, i.e., Steps (4) and (5) in Section 4.3. The reason is

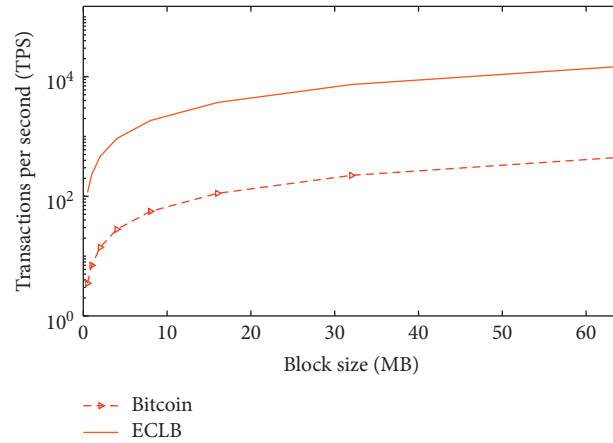


FIGURE 9: Tx throughput with different block sizes.

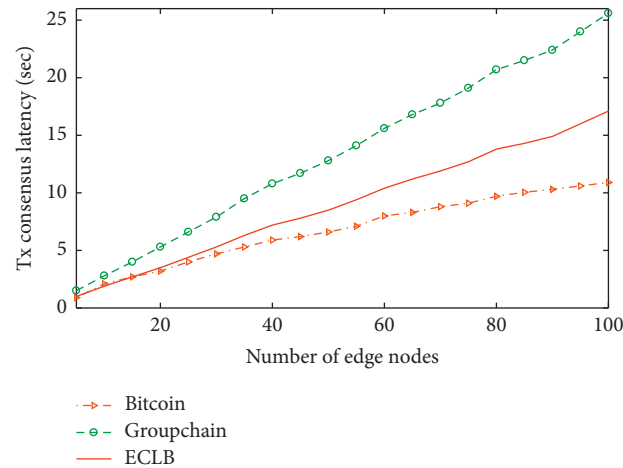


FIGURE 10: Transaction consensus latency with different number of edge nodes.

that the other 3 steps need to run only once and can be performed offline and in advance. Figure 11 shows the online encryption and decryption time cost. We observe that they are constant and low enough to satisfy efficient transaction regulation.

7. Related Works

In this section, we introduce some related works in the area of lightweight blockchain and access control.

7.1. Lightweight Blockchain. Since the advent of blockchain technology, much effort has been devoted to designing lightweight blockchain systems for decentralized Internet of Things [25]. Liu et al. [15] proposed a lightweight blockchain system to alleviate the resource occupation of blockchain and made it suitable for IIoT. Specifically, the work exploited an Unrelated Block Offloading Filter (UBOF) to detect and

offload unrelated transactions, thus achieving lightweight feature. However, offloading “unrelated transactions” will hinder the transaction regulation in the future. For long-term consideration, all the transaction records should be stored completely. Qu et al. [26] proposed a lightweight blockchain model based on hypergraphs. They used the hypergraph theory to partition the entire network into many hyperedges. Each hyperedge stores a part of transaction data to reduce the storage pressure. However, there are many nodes thus many transaction copies inside the same hyperedge, and one node might belong to more than one hyperedge. But, it brings inconvenience for transaction data sharing. In addition, high data redundancy is still not well-addressed. Cebe et al. [10] proposed an integrated lightweight blockchain framework for forensic applications of connected vehicles. In the work, each participant maintains a shared ledger and a fragmented ledger. The shared ledger keeps only hash values. The fragmented ledger keeps only some information that is of interest to the corresponding

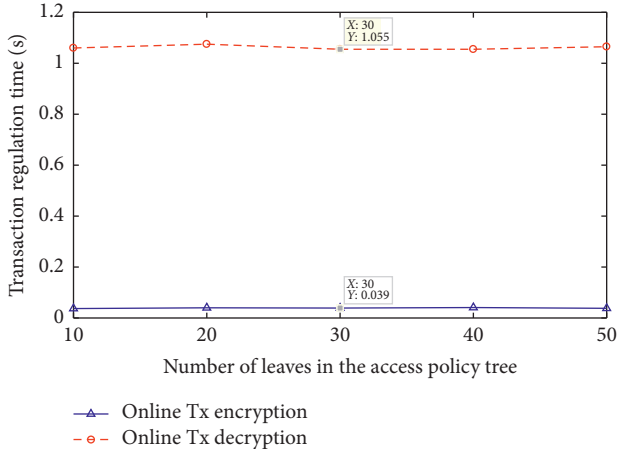


FIGURE 11: Transaction regulation efficiency with different leaf numbers of an access policy tree.

participants. The fragmented ledger greatly inspired us to design a lightweight ledger storage format at weak end devices. Lei et al. [13] proposed Groupchain, a novel scalable public blockchain of a two-chain structure suitable for fog computing of IoT services computing. Groupchain designed a lightweight transaction confirmation protocol to realize 0-block confirmation.

There are also many other works on lightweight blockchain [15–17, 27–29]. Nevertheless, all these works do not achieve light weight in terms of both ledger storage and block mining computation. A simple comparison is shown in Table 5.

7.2. Access Control. In this section, we will discuss some related works where the access control mechanisms were designed to achieve both privacy preservation and flexible data sharing.

Identity-based encryption enables fine-grained data access control [30–32]. As an advancement, attribute-based encryption (ABE) defines a user identity by his/her attribute set. Sahai and Waters [33] first proposed this method to exert access control over encrypted data. Later, Goyal et al. [34] extended the ABE method to key-policy attribute-based encryption (KP-ABE), by associating a user’s secret key with an access policy over attributes. The user can decrypt the ciphertext if and only if the attribute set of the ciphertext satisfies the access policy specified in his/her secret key. The encryptor exerts no control over who has access to the data being encrypted. Bethencourt et al. [18] extended the ABE method to the ciphertext-policy attribute-based encryption (CP-ABE), by associating the ciphertext with an access policy over attributes. A user’s secret attribute key is generated from his identity attribute set. The user can decrypt the ciphertext if and only if his/her attribute set satisfies the access policy specified in the ciphertext. The access policy maker is able to decide who should have access to the encrypted data. Currently, many works have been done to devote the ABE method to outsourcing and sharing data securely and flexibly.

TABLE 5: The comparison of lightweight properties in some lightweight blockchain systems.

Works	Lightweight ledger storage	Lightweight block mining
[10]	✓	✗
[13]	✗	✓
[15]	✓	✗
[16]	✓	✗
[17]	✓	✗
[26]	✓	✗
[27]	✗	✓
[28]	✗	✓
[29]	✓	✗
Our ECLB	✓	✓

Ding et al. [35, 36] proposed a privacy-preserving data processing scheme with flexible access control based on the homomorphic encryption of ABE. It realizes various computations over encrypted data in an efficient way and simultaneously flexibly controls the access to data processing results. Belguith et al. [37] introduced a securely outsourcing multi-authority ABE scheme with policy hidden for the cloud-assisted IoT. Our another work [38] proposed an efficient fine-grained access precision control (FAPC) scheme to achieve secure sharing of the same data, under different precisions with different data users. Deng et al. [20] combined the identity-based encryption and identity-based broadcast encryption mechanisms to propose an identity-based encryption transformation scheme. It supports the encrypted data shared with more people beyond those initially designated by the data owner. Xiong et al. [39] constructed a CP-ABE-based storage model for data storing and secure access in a cloud for IoT applications. It introduces an attribute authority management (AAM) module in the cloud storage system functioning as an agent that provides a user-friendly access control and highly reduces the storage overhead of public keys. Multiple ABE approaches have been proposed to implement secure data outsourcing [40–43], and keyword searching [44–46].

Considering the real-time requirement for transaction regulation, we combine the CP-ABE with the key encapsulation mechanism, to design an efficient transaction regulation protocol.

8. Discussion of Forks

The fork problems are not discussed above. There are two parallel chains in our ECLB, i.e., the leader chain and the Tx chain. Hence, there are two kinds of forks. Now, we will talk about the corresponding solutions, respectively.

- (1) The leader chain fork: it is the first important problem to solve, since it is the leader who guarantees the security of Tx blocks. Here, we will employ the corresponding solution in [13]. For ease of reading, we now recap it. Assume that there are k conflicted leader blocks lb_i^j , where $i \in \{0, 1, \dots, k-1\}$. Each edge node concatenates

these k block header hash strings $H(lb_i^k)$ in a uniform order (e.g., from low to high) as

$$\text{Hash} \leftarrow H\left(H(lb_t^1) \parallel H(lb_t^2) \parallel \dots \parallel H(lb_t^k)\right). \quad (12)$$

Then, the final winner leader block is

$$i = \text{Hash.substring}(0, k-1) \bmod k \quad (13)$$

- (2) The Tx chain fork: assume that N_i is the previous leader node, Node $_j$ the new leader node packed by a new leader block lb_t , t the time stamp of this new leader block. After the generation of this new leader block lb_t , the previous leader node N_i and some other edge nodes might receive lb_t with some delay, due to the bad network. Thus, N_i still keeps on packing and broadcasting some Tx blocks. Assume that these Tx blocks are denoted as $BS_{N_i} = \{b_{t_1}, b_{t_2}, \dots, b_{t_m}\}$, simultaneously, the new leader Node $_j$ also is packing and broadcasting some Tx blocks. As a consequence, there will be a time overlap between BS_{N_i} and the Tx blocks by the new leader Node $_j$. It will cause chaos of the transaction verification. Our solution to this Tx chain fork is to set only the Tx blocks with time stamps no later than t valid and remained in the Tx chain, namely $\{b_x | b_x \in BS_{N_i}, x \leq t\}$. Otherwise, the Tx blocks packed by the previous leader but with time stamps later than t will be all abandoned.

9. Conclusions

In this paper, we propose an edge-computing-based lightweight blockchain (ECLB) framework for mobile systems. In the ECLB framework, the edge nodes play a minor role. As a consequence, the storage and computation pressure at end mobile devices are greatly relieved. The fragmented ledger is employed as the storage format at end mobile devices. In this way, the end mobile devices not only can obtain information of interest but also do not need to store an entire copy of the ledger. Moreover, we design a two-chain structure of a leader chain and a transaction chain. These two chains grow in parallel. It greatly improves the throughput and confirmation latency. In addition, considering the regulation requirements under the permissioned blockchain setting, we specifically design a pluggable, secure, and efficient transaction regulation protocol. Finally, we give some formal security analysis and performance evaluation. It is demonstrated that our ECLB framework is secure and feasible.

Data Availability

All the experimental data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that there are no conflicts of interest.

Acknowledgments

This work was supported by the National Key R&D Program of China (grant no. 2020YFB1005500), the National Natural Science Foundation of China (grant numbers 62002139, U1736216, and 61902157), the Natural Science Foundation of Jiangsu Province (grant numbers BK20200886 and BK20200888), and the Project funded by China Postdoctoral Science Foundation (grant numbers 2019M651738 and 2019M661753).

References

- [1] S. Nakamoto, "Bitcoin: a peer-to-peer electronic cash system," *SSRN Electronic Journal*, 2008.
- [2] "ANTMINER S9 Hydro Miner," 2018, <https://www.bitmain.com/>.
- [3] "Blockchain Size," 2020, <https://bitinfocharts.com/>.
- [4] B. Vitalik, "Ethereum White Paper: A Next Generation Smart Contract & Decentralized Application Platform," 2013, <http://Ethereum.org>.
- [5] M. A. Javarone and C. S. Wright, "From Bitcoin to Bitcoin cash: a network analysis," *Proceedings of the 1st Workshop on Cryptocurrencies and Blockchains for Distributed Systems*, pp. 77–81, ACM, New York, NY, USA, 2018.
- [6] "Litecoin: Open Source P2p Internet Currency," 2011, <https://litecoin.org/>.
- [7] H. Wu, L. Wang, and G. Xue, "Privacy-aware task allocation and data aggregation in fog-assisted spatial crowdsourcing," *IEEE Transactions on Network Science and Engineering*, vol. 7, no. 1, pp. 589–602, 2020.
- [8] J. Zhang, H. Zhong, J. Cui, M. Tian, Y. Xu, and L. Liu, "Edge computing-based privacy-preserving authentication framework and protocol for 5G-enabled vehicular networks," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 7, pp. 7940–7954, 2020.
- [9] N. Abbas, Y. Zhang, A. Taherkordi, and T. Skeie, "Mobile edge computing: a survey," *IEEE Internet of Things Journal*, vol. 5, no. 1, pp. 450–465, 2018.
- [10] M. Cebe, E. Erdin, K. Akkaya, H. Aksu, and S. Uluagac, "Block4Forensic: an integrated lightweight blockchain framework for forensics applications of connected vehicles," *IEEE Communications Magazine*, vol. 56, no. 10, pp. 50–57, 2018.
- [11] M. Liu, F. R. Yu, Y. Teng, V. C. M. Leung, and M. Song, "Computation offloading and content caching in wireless blockchain networks with mobile edge computing," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 11, pp. 11008–11021, 2018.
- [12] W. Chen, Z. Zhang, Z. Hong et al., "Cooperative and distributed computation offloading for blockchain-empowered industrial internet of things," *IEEE Internet of Things Journal*, vol. 6, no. 5, pp. 8433–8446, 2019.
- [13] K. Lei, M. Du, J. Huang, and T. Jin, "Groupchain: towards a scalable public blockchain in fog computing of IoT services computing," *IEEE Transactions on Services Computing*, vol. 13, no. 2, pp. 252–262, 2020.
- [14] I. Eyal, A. E. Gencer, E. G. Sirer, and R. Van Renesse, "Bitcoin-NG: a scalable blockchain protocol," in *Proceedings of the 13th Usenix Conference on Networked Systems Design and Implementation (NSDI)*, pp. 45–59, USENIX Association, Boston, MA, USA, April 2016.

- [15] Y. Liu, K. Wang, Y. Lin, and W. Xu, “ $\mathit{LightChain}$: a lightweight blockchain system for industrial internet of things,” *IEEE Transactions on Industrial Informatics*, vol. 15, no. 6, pp. 3571–3581, 2019.
- [16] M. Zamani, M. Movahedi, and M. Raykova, “RapidChain: Scaling blockchain via full sharding,” in *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security (CCS)*, pp. 931–948, ACM, New York, NY, USA, 2018.
- [17] E. Kokoris-Kogias, P. Jovanovic, L. Gasser, N. Gailly, E. Syta, and B. Ford, “Omniledger: a secure, scale-out, decentralized ledger via sharding,” in *Proceedings of the IEEE Symposium on Security and Privacy (SP)*, pp. 583–598, San Francisco, CA, USA, May 2018.
- [18] J. Bethencourt, A. Sahai, and B. Waters, “Ciphertext-policy attribute-based encryption,” in *Proceedings of the IEEE Symposium on Security and Privacy (SP ’07)*, pp. 321–334, Berkeley, CA, USA, 2007.
- [19] J. Katz and Y. Lindell, *Introduction to Modern Cryptography*, Chapman & Hall/CRC, London, UK, 2nd edition, 2014.
- [20] H. Deng, Z. Qin, Q. Wu et al., “Identity-based encryption transformation for flexible sharing of encrypted data in public cloud,” *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 3168–3180, 2020.
- [21] J. R. Douceur, “*The Sybil Attack*,” *Peer-To-Peer Systems*, P. Druschel, F. Kaashoek, and A. Rowstron, Eds., pp. 251–260, Springer, Berlin, Germany, 2002.
- [22] G. O. Karame, E. Androulaki, and S. Capkun, “Double-spending fast payments in Bitcoin,” in *Proceedings of the 2012 ACM Conference on Computer and Communications Security (CCS)*, pp. 906–917, Association for Computing Machinery, New York, NY, USA, 2012.
- [23] A. Gervais, G. Karame, K. Wst, V. Glykantzis, H. Ritzdorf, and S. Capkun, “On the security and performance of proof of work blockchains,” in *Proceedings of the 23rd ACM SIGSAC Conference on Computer and Communication Security (CCS)*, ACM, Vienna, Austria, October 2016.
- [24] A. De Caro and V. Iovino, “JPBC: Java pairing based cryptography,” in *Proceedings of the IEEE Symposium on Computers and Communications (ISCC)*, pp. 850–855, Kerkyra, Greece, July 2011.
- [25] Y. Xu, J. Liu, Y. Shen, J. Liu, X. Jiang, and T. Taleb, “Incentive jamming-based secure routing in decentralized internet of things,” *IEEE Internet of Things Journal*, vol. 8, no. 4, pp. 3000–3013, 2021.
- [26] C. Qu, M. Tao, and R. Yuan, “A hypergraph-based blockchain model and application in internet of things-enabled smart homes,” *Sensors*, vol. 18, no. 9, p. 2784, 2018.
- [27] E. Gutierrez, T. P. Monath, A. Alava, D. Uriguen, M. Arzube, and R. W Chamberlain, “Epidemiologic investigations of the 1969 epidemic of Venezuelan encephalitis in Ecuador,” *American Journal of Epidemiology*, vol. 102, no. 5, 1975.
- [28] A. Dorri, S. S. Kanhere, and R. Jurdak, “Towards an optimized blockchain for IoT,” in *Proceedings of the Second International Conference on Internet-of-Things Design and Implementation (IoTDI)*, pp. 173–178, ACM, New York, NY, USA, April 2017.
- [29] K. Karlsson, “Vegvisor: a partition-tolerant blockchain for the internet-of-things,” in *Proceedings of the 2018 IEEE 38th International Conference on Distributed Computing Systems (ICDCS)*, pp. 1150–1158, Vienna, Austria, July 2018.
- [30] D. Ferraiolo, J. Cugini, and D. R. Kuhn, “Role-based access control (rbac): features and motivations,” in *Proceedings of 11th Annual Computer Security Application Conference*, pp. 241–248, New Orleans, LA, USA, December 1995.
- [31] R. S. Sandhu, E. J. Coyne, H. L. Feinstein, and C. E. Youman, “Role-based access control models,” *Computer*, vol. 29, no. 2, pp. 38–47, 1996.
- [32] D. F. Ferraiolo, R. Sandhu, S. Gavrila, D. R. Kuhn, and R. Chandramouli, “Proposed NIST standard for role-based access control,” *ACM Transactions on Information and System Security*, vol. 4, no. 3, pp. 224–274, 2001.
- [33] A. Sahai and B. Waters, “Fuzzy identity-based encryption,” in *Proceedings of the Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 457–473, Springer, Aarhus, Denmark, May 2005.
- [34] V. Goyal, O. Pandey, A. Sahai, and B. Waters, “Attribute-based encryption for fine-grained access control of encrypted data,” in *Proceedings of the 13th ACM Conference on Computer and Communications Security (CCS)*, pp. 89–98, ACM, Alexandria, VA, USA, October 2006.
- [35] W. Ding, Z. Yan, and R. H. Deng, “Privacy-preserving data processing with flexible access control,” *IEEE Transactions on Dependable and Secure Computing*, vol. 17, no. 2, pp. 363–376, 2020.
- [36] Z. Brakerski, D. Cash, R. Tsabary, and H. Wee, “Targeted homomorphic attribute-based encryption,” in *Proceedings of the Theory of Cryptography Conference*, pp. 330–360, Springer, Beijing, China, November 2016.
- [37] S. Belguith, N. Kaaniche, M. Laurent, A. Jemai, and R. Attia, “PHOABE: securely outsourcing multi-authority attribute based encryption with policy hidden for cloud assisted IoT,” *Computer Networks*, vol. 133, pp. 141–156, 2018.
- [38] E. Matusik, T. P. Gibson, K. Cheng, G. G. Dagher, L. Wang, and S. Yu, “Fluorometric assay for N-acetylprocainamide,” *Clinical Chemistry*, vol. 21, no. 13, pp. 1899–1902, 1975.
- [39] S. Xiong, Q. Ni, L. Wang, and Q. Wang, “SEM-ACSIT: secure and efficient multiauthority access control for IoT cloud storage,” *IEEE Internet of Things Journal*, vol. 7, no. 4, pp. 2914–2927, 2020.
- [40] H. Zhong, Y. Zhou, Q. Zhang, Y. Xu, and J. Cui, “An efficient and outsourcing-supported attribute-based access control scheme for edge-enabled smart healthcare,” *Future Generation Computer Systems*, vol. 115, pp. 486–496, 2021.
- [41] W. C. Garrison, A. Shull, S. Myers, and A. J. Lee, “On the practicality of cryptographically enforcing dynamic access control policies in the cloud,” in *Proceedings of the IEEE Symposium on Security and Privacy (SP)*, pp. 819–838, San Jose, CA, USA, May 2016.
- [42] M. Li, S. Yu, Y. Zheng, K. Ren, and W. Lou, “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.
- [43] T. Jung, X.-Y. Li, Z. Wan, and M. Wan, “Control cloud data access privilege and anonymity with fully anonymous attribute-based encryption,” *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 1, pp. 190–199, 2015.
- [44] W. Sun, S. Yu, W. Lou, Y. T. Hou, and H. Li, “Protecting your right: verifiable attribute-based keyword search with fine-grained owner-enforced search authorization in the cloud,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 27, no. 4, pp. 1187–1198, 2016.
- [45] K. He, J. Guo, J. Weng, J. K. Liu, and X. Yi, “Attribute-based hybrid boolean keyword search over outsourced encrypted data,” *IEEE Transactions on Dependable and Secure Computing*, vol. 17, no. 6, pp. 1207–1217, 2020.
- [46] H. Li, Y. Yang, T. H. Luan, X. Liang, L. Zhou, and X. S. Shen, “Enabling fine-grained multi-keyword search supporting classified sub-dictionaries over encrypted cloud data,” *IEEE Transactions on Dependable and Secure Computing*, vol. 13, no. 3, pp. 312–325, 2016.