IJNS
International Journal of Network Security

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

Vol. 20, No. 4 (July 2018)

INTERNATIONAL JOURNAL OF NETWORK SECURITY

Editor-in-Chief

Prof. Min-Shiang Hwang
Department of Computer Science & Information Engineering, Asia University, Taiwan

Co-Editor-in-Chief:

Prof. Chin-Chen Chang (IEEE Fellow)
Department of Information Engineering and Computer Science, Feng Chia University, Taiwan

Publishing Editors

Shu-Fen Chiou, Chia-Chun Wu, Cheng-Yi Yang

Board of Editors

Ajith Abraham
School of Computer Science and Engineering, Chung-Ang University (Korea)

Wael Adi
Institute for Computer and Communication Network Engineering, Technical University of Braunschweig (Germany)

Sheikh Iqbal Ahamed
Department of Math., Stat. and Computer Sc. Marquette University, Milwaukee (USA)

Vijay Atluri
MSIS Department Research Director, CIMIC Rutgers University (USA)

Mauro Barni
Dipartimento di Ingegneria dell'Informazione, Università di Siena (Italy)

Andrew Blyth
Information Security Research Group, School of Computing, University of Glamorgan (UK)

Soon Ae Chun
College of Staten Island, City University of New York, Staten Island, NY (USA)

Stefanos Gritzalis
University of the Aegean (Greece)

Lakhmi Jain
School of Electrical and Information Engineering, University of South Australia (Australia)

James B D Joshi
Dept. of Information Science and Telecommunications, University of Pittsburgh (USA)

Ç etin Kaya Koç
School of EECS, Oregon State University (USA)

Shahram Latifi
Department of Electrical and Computer Engineering, University of Nevada, Las Vegas (USA)

Cheng-Chi Lee
Department of Library and Information Science, Fu Jen Catholic University (Taiwan)

Chun-Ta Li
Department of Information Management, Tainan University of Technology (Taiwan)

Iuon-Chang Lin
Department of Management of Information Systems, National Chung Hsing University (Taiwan)

John C.S. Lui
Department of Computer Science & Engineering, Chinese University of Hong Kong (Hong Kong)

Kia Makki
Telecommunications and Information Technology Institute, College of Engineering, Florida International University (USA)

Gregorio Martinez
University of Murcia (UMU) (Spain)

Sabah M.A. Mohammed
Department of Computer Science, Lakehead University (Canada)

Lakshmi Narasimhan
School of Electrical Engineering and Computer Science, University of Newcastle (Australia)

Khaled E. A. Negm
Etisalat University College (United Arab Emirates)

Joon S. Park
School of Information Studies, Syracuse University (USA)

Antonio Pescapè
University of Napoli "Federico II" (Italy)

Zuhua Shao
Department of Computer and Electronic Engineering, Zhejiang University of Science and Technology (China)

Mukesh Singhal
Department of Computer Science, University of Kentucky (USA)

Nicolas Sklavos
Informatics & MM Department, Technological Educational Institute of Patras, Hellas (Greece)

Tony Thomas
School of Computer Engineering, Nanyang Technological University (Singapore)

Mohsen Toorani
Department of Informatics, University of Bergen (Norway)

Shuozhong Wang
School of Communication and Information Engineering, Shanghai University (China)

Zhi-Hui Wang
School of Software, Dalian University of Technology (China)

Chuan-Kun Wu
Chinese Academy of Sciences (P.R. China) and Department of Computer Science, National Australian University (Australia)

Chou-Chen Yang
Department of Management of Information Systems, National Chung Hsing University (Taiwan)

Sherali Zeadally
Department of Computer Science and Information Technology, University of the District of Columbia, USA

Jianping Zeng
School of Computer Science, Fudan University (China)

Justin Zhan
School of Information Technology & Engineering, University of Ottawa (Canada)

Mingwu Zhang
College of Information, South China Agric University (China)

Yan Zhang
Wireless Communications Laboratory, NICT (Singapore)

PUBLISHING OFFICE

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

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

PUBLISHER: Candy C. H. Lin

© Jalaxy Technology Co., Ltd., Taiwan 2005
23-75, P.O. Box, Taichung, Taiwan 40199, R.O.C.

International Journal of Network Security

Vol. 20, No. 4 (July 1, 2018)

1. Security Analysis of Discrete Event Based Threat Driven Authentication Approach in VANET Using Petri Nets
Arun Malik, Babita Pandey 601-608

2. An Image Retrieval Algorithm Based on Gist and Sift Features
Bei Xie, Jiaohua Qin, Xuyu Xiang, Hao Li and Lili Pan 609-616

3. Diffie-Hellman Type Key Exchange, ElGamal Like Encryption/Decryption and Proxy Re-encryption Using Circulant Matrices
Chitra Rajarama, Jagadeesha Narasimhamurthy Sugatoor, and T. Yerri Swamy 617-624

4. Performance Analysis of RSA and Elliptic Curve Cryptography
Dindayal Mahto and Dilip Kumar Yadav 625-635

5. Security Improvements of EPS-AKA Protocol
Mourad Abdeljebbar and Rachid El Kouch 636-644

6. Benchmark Datasets for Network Intrusion Detection: A Review
Yasir Hamid, V. R. Balasaraswathi, Ludovic Journaux and M. Sugumaran 645-654

7. An Efficient Practice of Privacy Implementation: Kerberos and Markov Chain to Secure File Transfer Sessions
Fadi Al-Ayed, Chunqiang Hu and Hang Liu 655-663

8. Research on Cloud Computing Security Risk Assessment Based on Information Entropy and Markov Chain
Ming Yang, Rong Jiang, Tilei Gao, Wanyu Xie and Jia Wang 664-673

9. Study on Trust Model for Multi-users in Cloud Computing
Xu Wu 674-682

10. Analysis of One Dynamic Multi-Keyword Ranked Search Scheme over Encrypted Cloud Data
Zhengjun Cao, Chong Mao, Lihua Liu, Wenping Kong and Jinbo Wang 683-688

11. CP-ABE for Selective Access with Scalable Revocation: A Case Study for Mobile-based Healthfolder
Divyashikha Sethia, Huzur Saran and Daya Gupta 689-701

12. Data Verification Using Block Level Batch Auditing on Multi-cloud Server
G. L. Prakash, Manish Prateek and Inder Singh 702-709

13. A New Access Control System Based on CP-ABE in Named Data Networking
Tao Feng and Jiaqi Guo 710-720

14. Defense Techniques of SYN Flood Attack Characterization and Comparisons
Shaila Ghanti and G. M. Naik 721-729

15. Efficient Anomaly Intrusion Detection Using Hybrid Probabilistic Techniques in Wireless Ad Hoc Network
K. Murugan and P. Suresh 730-737

16. Identification of Cyber Criminal by Analysing the Users Profile
K. Veena and K. Meena 738-745

17. An Outlook on Cryptographic and Trust Methodologies for Clusters Based Security in Mobile Ad Hoc Networks
V. S. Janani and M. S. K. Manikandan 746-753

18. A Credible Mechanism of Service about Data Resource in Cloud Computing
Yunfa Li, Yangyang Shen and Mingyi Li 754-761

19. Traceable Certificateless Ring Signature Scheme For No Full Anonymous Applications
Ke Gu, LinYu Wang, Na Wu and NianDong Liao 762-773

20. An Untraceable Voting Scheme Based on Pairs of Signatures
Kazi Md. Rokibul Alam, Adnan Maruf, Md. Rezaur Rahman Rakib, G. G. Md. Nawaz Ali, Peter Han Joo Chong, and Yasuhiko Morimoto 774-787

21. Multiple New Formulas for Cipher Performance Computing
Youssef Harmouch, Rachid Elkouch and Hussain Ben-Azza 788-800

22. A Novel Dual Image-based High Payload Reversible Hiding Technique Using LSB Matching
Yu-Lun Wang, Jau-Ji Shen, and Min-Shiang Hwang 801-804

23. Defense of Computer Network Viruses Based on Data Mining Technology
Cen Zuo 805-810

Security Analysis of Discrete Event Based Threat Driven Authentication Approach in VANET Using Petri Nets

Arun Malik[1] and Babita Pandey[2]
(Corresponding author: Babita Pandey)
Department of Computer Science and Engineering, Lovely Professional University[1]
Department of Computer Applications, Lovely Professional University[2]
Jalandhar - Delhi G. T. Road, Phagwara, Punjab 144411, India
(Email: arunmalikhisar@gmail.com)
(Received Nov. 3, 2016; revised and submitted Mar. 31, 2017)

Abstract

Vehicular Ad hoc Network (VANET) is identified as a key part of Intelligent Transport framework. VANET plays a significant role to establish communication between Vehicle to Vehicle (V2V) and Vehicle to Infrastructure (V2I). Keeping in mind the end goal to build an effective network, it is expected to have steadiness of security and transmission of unwavering quality in VANET. In this paper, a discrete event based threat driven authentication approach to provide secure communication between V2V and V2I is proposed. A combination of re-encryption key, public key, private key and session key is used by this approach for guaranteeing a secure communication between vehicle to vehicle and vehicle to Road side Unit (RSU). The analysis of the proposed approach is realized by using Petri nets and Veins framework. The proposed approach is also compared with the related works on the basis of computational overhead (CO), throughput, packet delivery ratio and average delay. The evaluated results reflect that proposed approach outperforms.

Keywords: Road Side Unit; Vehicle to Infrastructure; Vehicle to Vehicle; Vehicular Ad hoc Network

1 Introduction

VANET act as a promising application situation for facilitating intelligent transport system. In VANET, V2V and V2I are the two types of primary communications [6,7,17]. In VANET vehicles are having their communication devices, through the help of which vehicles initiate communication among themselves and also communicate with the RSUs placed at vital points on the side of road. In V2V communication, the on board unit(OBU) embedded in vehicles frequently broadcast information related to location of the vehicle, speed of the vehicle, present time, direction, acceleration/deceleration of the vehicles and traffic events [12, 18]. Due to which driver can obtain a better-quality understanding of their driving situations. Moreover, with the help of the information frequently broadcasted by OBU, VANET plays a vital role in prevention of accidents, in providing solution for heavy traffic jams and in broadcasting of alternate road messages [13]. On the other hand, vehicles are provided with internet facilities for accessing local information, entertainment, songs and movies in V2I communication [15].

Security and privacy are the major barricades in the successful deployment of VANET [9]. Before utilizing the above attractive applications of VANET, issues related to security and privacy must be resolved. Authentication and integrity of messages exchanged among vehicles and between vehicles and RSUs must be ensured. To provide security and privacy to the information exchanged among vehicles and between vehicles and RSUs, it is very important to establish secure authentication among vehicles and between vehicles and RSUs. Plenty of research work has been done in VANET related to privacy and security preservation based authentication schemes. But most of the existing authentication schemes are vulnerable to various types of attacks in VANET which results in lesser throughput and high computational cost Therefore, to design a secure authentication scheme with high throughput and low computational cost still remains a primary challenging problem in VANET.

Thus primary aim of this paper is to address the authentication problem among the vehicles and between vehicles and RSUs. The authentication among the vehicles and between vehicles and RSUs helps to prevent accidents and traffic jams. Moreover, it can also be utilized as fundamental information for any type of responsibility issue after accident. In order to provide an efficient security mechanism in VANET, this paper proposes a discrete event based threat driven authentication approach. Per-

formance of the proposed approach is analyzed by using Petri net model and veins framework.

The remainder of this paper is organized as follow. Section 2 describes the related work. Next, Section 3 describes the methodology of proposed authentication approach. Section 4 describes the petri net model of the proposed authentication approach. Section 5 describes the system model used to simulate the proposed authentication approach. Results and discussions are described in Section 6 before concluding the paper in section 7.

2 Related Works

This section describes in brief the different types of existing authentication schemes in VANET.

In [10] to diminish the entrust of authentication delay, an authentication scheme is proposed that utilize a process based on dynamic session secret to increase the computational efficiency and speed of authentication procedure. This scheme has efficient authentication ability and safeguards the VANET towards variety of malicious attacks.

In [11] an efficient authentication protocol is proposed to provide an anonymous authentication that utilize certificate less signcryption without pairing. Even in the absence of RSU, the proposed protocol performs efficiently.

In [14] an exhaustive message authentication scheme is proposed that provide authentication among inter RSU ranges and between Intra RSU ranges. This authentication scheme also permits the hand off among different RSUs. This scheme provides an efficient secure communications by balancing the computational overhead.

In [4] a trust evaluation mechanism is utilized to calculate trust value for the nodes. The trust values assigned to nodes are useful in identifying the malicious nodes in the network. Moreover, a layered structure is demonstrated to establish communication among the authenticating vehicles.

In [5] a novel and efficient authentication method is proposed to provide authentication to public and private vehicles in VANET. To speed up the process of many to one communication, the total signature and total signcryption are utilized.

In [16] an authentication protocol based on blind signature scheme is proposed for I2V communication. This authentication scheme not only provide speedy authentication but also assure the security and location anonymity to the vehicle.

In [2] an authentication scheme is described that utilize identity based signature mechanism in a way to provide multiple level of secrecy to vehicles in VANET. This authentication scheme utilizes an efficient pseudonyms issuance mechanism with the help of which pseudonyms issuer can issue unique pseudonyms to the vehicles. Moreover, each pseudonym binds with the expiration date due to which no public key certificate is required by this protocol to implement short term credentials.

In [1] time stamp based authentication approach is proposed to provide authentication among vehicles and RSU. Legal users are protected from malicious attacks with the help of this authentication approach. This authentication approach provides privacy to every vehicle by not revealing the original identity of vehicles.

In [3] a light weight authentication scheme is mentioned that provides authentication among vehicles and RSUs in VANET. This scheme utilizes hash function, XOR operation and symmetric cryptography to provide privacy and security among vehicle and RSU in VANET.

In [8] a novel ID based authentication scheme is proposed to provide secure RSU to Vehicle communication in VANET. For authentication this scheme uses road pass ticket and vehicle plate number. The effectiveness of this scheme is is analyzed by using Petri nets.

The aforementioned conventional authentication schemes discussed so far do not provide the complete authentication solution for VANET. As these schemes either provides authentication among the vehicles or between vehicles and RSUs. None of the aforementioned authentication schemes provides authentication among the vehicles and between vehicles and RSUs together which results in lesser throughput and high computational overhead. Therefore, the authentication approach proposed in this paper provides authentication among the vehicles as well as the authentication between vehicles and RSUs. Thus providing the complete authentication solution for VANET.

3 Proposed Authentication Approach

In this approach, before starting the authentication process among vehicles and between vehicles and RSUs, credential provider provides the credentials related to Vehicle and RSU that enter into the VANET. Credentials includes public key of moving vehicle, private key of moving vehicle, session key, public key of credential provider, re-encryption key of moving vehicle, public key of fixed RSU, private key of fixed RSU, re-encryption key of fixed RSU.

Step by step procedure used for the authentication between vehicle and RSU is as follow:

- Vehicle sends a message containing arbitrary number X1, particular time instance t1, and session key S1. The sent message is first encrypted with public key of fixed RSU that can be decrypted by private key of RSU.

- RSU initiates message transmission to vehicle containing arbitrary number X1 send by vehicle, arbitrary number X2 generated by RSU, Session key S1 and a particular time instance t2. Then, message is encrypted with the public key of credential provider.

- Public key of credential provider plus re-encrypt key of moving vehicle generates public key of moving vehicle. When public key of vehicle is next generated the message is decrypted by the private key of moving vehicle. Arbitrary number X1 generated by vehicle is verified.
- Vehicle sends a message to RSU containing arbitrary number X2 generated by RSU, Session key S1 and a particular time instance t3. The message is then encrypted with the public key of credential provider
- Public key of credential provider plus re-encrypt key of RSU generates public key of fixed RSU. When public key of fixed RSU is generated the message is then decrypted with the private key of fixed RSU. Arbitrary number generated by the RSU is verified.
- Authentication is over after verifying the arbitrary number X1 and X2 by both vehicle and RSU and communication is established.

Step by step procedure used for the authentication between two vehicles is as follow:

- Vehicle V_i sends a message containing arbitrary number X1, particular time instance t1, and session key S1. The sent message is first encrypted with public key of Vehicle V_j that can be decrypted by private key of Vehicle 2.
- Vehicle V_j initiates message transmission to vehicle V_i containing arbitrary number X1 send by vehicle, arbitrary number X2 generated by vehicle V_j, Session key S1 and a particular time instance t2. Then, message is encrypted with the public key of credential provider.
- Public key of credential provider plus re-encrypt key of vehicle V_i generates public key of vehicle V_i. When public key of vehicle V_i is generated the message is decrypted by the private key of vehicle V_i. Arbitrary number X1 generated by vehicle V_i is verified.
- Vehicle V_i sends a message to vehicle V_j containing arbitrary number X2 generated by vehicle V_j, Session key S1 and a particular time instance t3. The message is then encrypted with the public key of credential provider
- Public key of credential provider plus re-encrypt key of vehicle V_j generates public key of vehicle V_j. When public key of vehicle V_j is generated the message is then decrypted with the private key of vehicle V_j. Arbitrary number generated by the vehicle V_j is verified.
- Authentication is over after verifying the arbitrary number X1 and X2 by both vehicle V_i and vehicle V_j and communication is established.

Algorithms for establishing mutual authentication are listed in Algorithms 1 and 2.

Algorithm 1 Authentication between Vehicle and RSU

Begin
Initialize the *authentication.* .
while Authentication session not end **do**
  Vehicle send message containing*(X1,S1, t1)* encrypted with public key of RSU
  RSU Decrypts the message by its private key.
  RSU then send the message containing *(X1,X2,S1, t2)* encrypted with public key of credential provider.
  Public key of Vehicle⇐*public key of credential provider + re-encryption key of vehicle*
  Vehicle decrypts the message by its private key
  if *X1* in the message send by RSU to the vehicle matches with the*X1* genrated by vehicle **then**
    X1 verified
  end if
  Vehicle send the message containing *(X2,S1, t3)* encrypted with public key of credential provider.
  Public key of RSU⇐*public key of credential provider + re-encryption key of RSU*
  RSU decrypts the message by its private key
  if *X2* in the message send by vehicle to the RSU matches with the*X2* genrated by the RSU **then**
    X2 verified
  end if
  Establish communication between Vehicle and RSU
end while
End

Algorithm 2 Authentication between Vehicle V_i and Vehicle V_j

Begin
Initialize the *authentication.* .
while Authentication session not end **do**
  V_i send message containing*(X1,S1, t1)* encrypted with public key of V_j
  V_j Decrypts the message by its private key.
  V_j then send the message containing *(X1,X2,S1, t2)* encrypted with public key of credential provider.
  Public key of V_i ⇐*public key of credential provider + re-encryption key of* V_i
  V_i decrypts the message by its private key
  if *X1* in the message send by V_j to the V_i matches with the*X1* genrated by V_i **then**
    X1 verified
  end if
  V_i send the message containing *(X2,S1, t3)* encrypted with public key of credential provider.
  Public key of V_j ⇐*public key of credential provider + re-encryption key of* V_j
  V_j decrypts the message by its private key
  if *X2* in the message send by V_i to the V_j matches with the*X2* genrated by the V_j **then**
    X2 verified
  end if
  Establish communication between V_i and V_j
end while
End

4 Petri Net Model for Proposed Authentication Approach

Petri net is widely used in depicting the dynamic behavior of system due to its simple and flexible nature. Petri net can be applied to variety of systems in the form of graphical and mathematical modeling tool. Petri net is capable tool to illustrate and learn information processing systems that are described as being synchronous, asynchronous, parallel, distributed, non deterministic, and/or stochastic. Petri net is utilized to draw flow charts, block diagrams and networks. Moreover, Petri net is used to simulate the lively and synchronized actions of the systems.

A discrete event based threat driven authentication approach for vehicles and RSUs has been proposed in the previous section. The proposed authentication approach is analyzed by using Petri net model that control arbitrary events and processes the input data. Petri net model carries out various types of token values from one place to another at the time of transition firings where initial marking is labeled by P0. The subsequent Petri net model and reachability graph for the proposed authentication approach is shown in Figures 1 and 2 respectively.

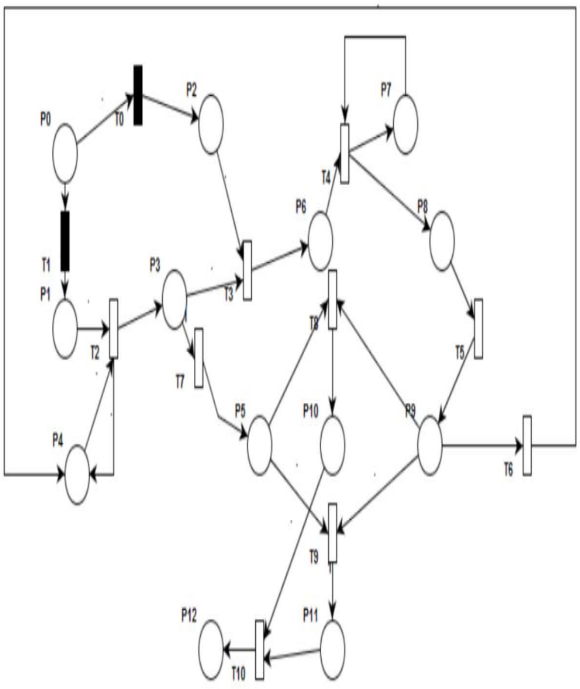


Figure 1: Petri net model for the proposed authentication approach

Reachability and liveness are the two important properties that must be possessed by the proposed authentication approach for the assessment of its correctness. Whether we can reach from one state to another is determined by the reachability. Whether all reachable state can be fired without coming into deadlock situation is determined by liveness. After testing the proposed authentication approach by using Petri net model, it was found that the proposed authentication approach possessed both reachability and liveness properties. Various types of marking and states that can be reached are depicted by reachability graph. In Figure 2 nodes represent markings and arrows are labeled with transition names to represent that markings are reached by firing a certain transitions.

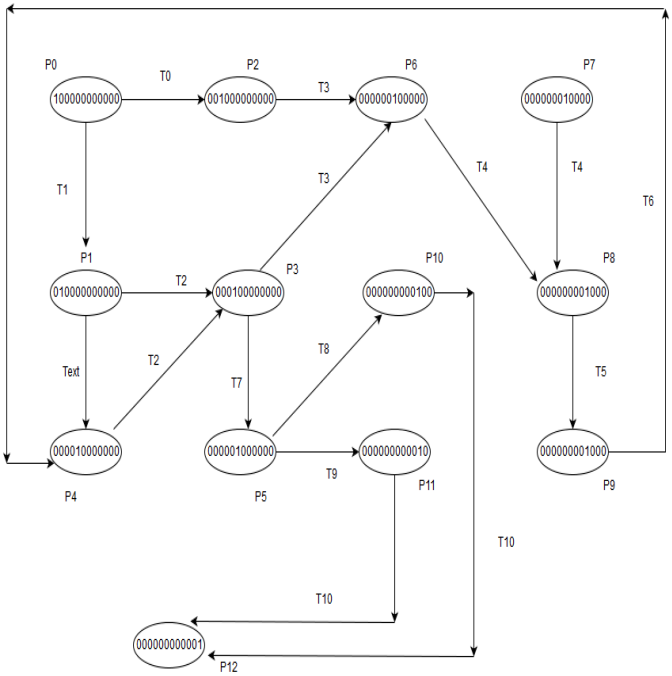


Figure 2: Reachability graph for proposed authentication approach

Table 1: Description of places

State	Description
P0	Working place of credential provider
P1	Original place of on board unit
P2	Original place of RSU
P3	Waiting place
P4	Working place of on board unit
P5	Keep on board unit information
P6	Waiting Place of RSU
P7	Working place of RSU
P8	Waiting place of RSU
P9	Keep information of RSU
P10	Verify information-Yes
P11	Verify information-No
P12	Workplace for authentication

For the better understanding of the Petri nets model for the proposed authentication approach, description of places and transitions used to represent the proposed authentication approach in Petri net model are shown in Table 1 and Table 1. To analyze the model of the proposed approach, Petri nets tool is utilized in Acer laptop with Window 7 environment. The methodology of the proposed model is carried out smoothly in the given simulation environment. The authentication process among vehicles and between vehicle and RSU is carried out initially whenever a vehicle enters the network. Different types of situations that RSUs and Vehicles can undergo during authentication process are identified from T0-T10 transitions.

Table 2: Description of transition

Transition	Description
T0	Receiving credentials of RSU
T1	Receiving credentials of Vehicle
T2	Process data received from vehicle
T3	Received Data from vehicle and RSU
T4	Process data received from RSU
T5	Received RSU data
T6	Process data received from RSU
T7	Received vehicle data
T8	Verify vehicle and RSU data-Yes
T9	Verify vehicle and RSU data-No
T10	data transmission for authentication

5 System Model

The proposed authentication approach is also compared with the authentication schemes mentioned in [1, 3, 8] by using vehicle in network simulation(Veins) framework. Veins is an open source framework suitable for VANET simulation. In Veins framework simulation model is executed with the help of OMNet++ which is an event based network simulator and SUMO which act as a road traffic simulator. The simulation parameters used to execute simulation moodel are mentioned in Tables 3 and 4.

6 Results and Discussions

The performance evaluation of the proposed authentication approach is compared with the existing authentication approach mentioned in [1, 3, 8] on the basis of CO, throughput, packet delivery ratio and average delay.

The excess time or indirect time required by the authentication approach to establish secure authentication among vehicles and between vehicles ans RSUs is termed as computational overhead (CO). Table 5 depicts the comparison of proposed authentication approach with the existing authentication approach mentioned in [1, 3, 8],

Table 3: Traffic simulation parameters

Parameter Name	Value
Number of Vehicles	5,10,15,20,25,30
Maximum Speed	22m/sec
Acceleration	$5m/s^2$
Deceleration	$8m/s^2$
Driver Fault	0.5

Table 4: Network simulation parameters

Parameter Name	Value
Network Simulator	OMNet++
Simulation Time	120 sec
Area of Simulation	400 meters x 400meters
Message Size	512 bytes
Simulation Set Up	Random and Cross roads
MAC Protocol	IEEE802.11p
Range of Transmission	300 meters

where RN represents cost of random number generation, HF represents cost of hash function, AE represents cost of executing asymmetric encryption using re-encryption key, SE represents cost of executing symmetric encryption, XF represents cost of executing XOR function.

Due to the use of complex mathematical function by asymmetric algorithms, they are considered to be the slower as compared to symmetric encryption algorithms. But with the existing state of computational technology, security of VANET depends mainly on the size of the keys of its encryption algorithms. With the advancement of technology both symmetric and asymmetric algorithms requires same key size. Moreover security of asymmetric encryption algorithm exists in the security of its private key that cannot be figure out from its public key. In addition to this, asymmetric encryption algorithm requires less number of secret keys as compared to symmetric algorithm. Asymmetric encryption algorithms are efficient for the encryption of short messages to provide security and privacy in VANET. As our proposed authentication approach utilizes asymmetric encryption algorithm and the other authentication approach mentioned in [1, 3, 8] uses symmetric algorithm, hash function and XOR function due to which CO of the proposed authentication approach is less as compared to the existing authentication approach mentioned in [1, 3, 8] as shown in Figure 3.

The number of data packets sent within a given time period over a physical or logical communication channel is termed as throughput. Throughput of the proposed authentication approach is better as compared to the exisitng authentication approach mentioned in [1, 3, 8] as shown in Figure 4.

The ratio of total number of data packets that are suceesfully received to the total number of data packets

Table 5: Comparison table

CO	Proposed Approach	Approach in [1]	Approach in [3]	Approach in [8]
RN	2	3	2	2
Hash Function	0	4	9	2
AE	2	0	0	0
SE	0	2	6	2
XF	0	3	2	2
Total Cost	2RN+2AE	3RN+4HF+2SE+3XF	2RN+9HF+6SE+2XF	2RN+2HF+2SE+2XF

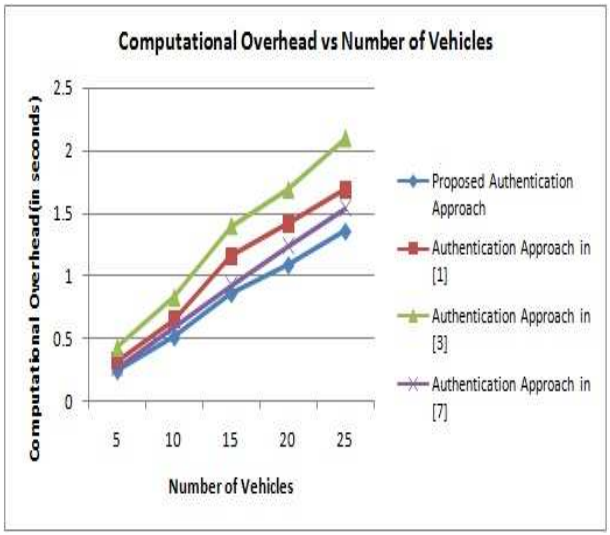


Figure 3: Computational overhead

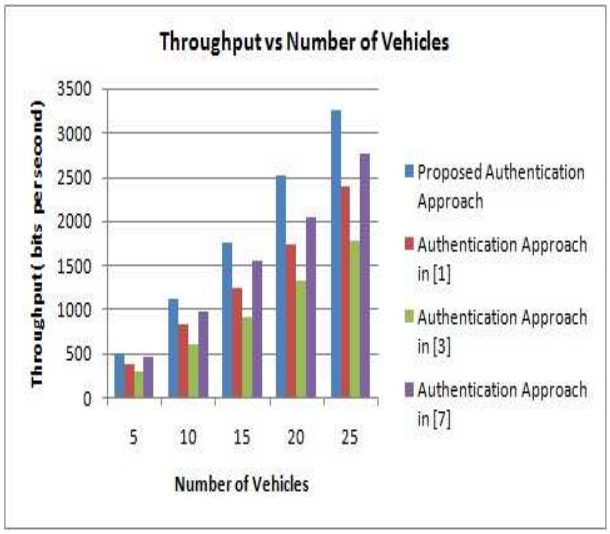


Figure 4: Throughput

sent is termed as packet delivery ratio.Packet delivery ratio of the proposed authentication approach is better as compared to the exisitng authentication approach mentioned in [1,3,8] as shown in Figure 5.

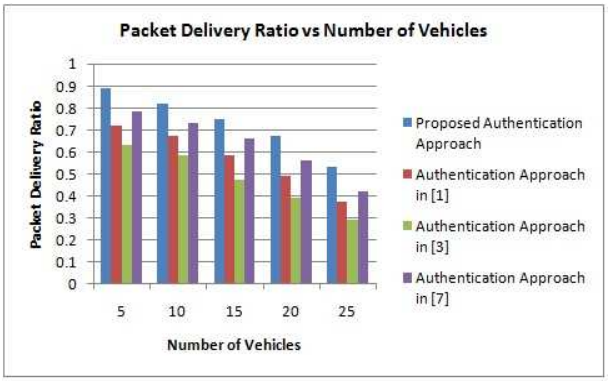


Figure 5: Packet delivery ratio

Time taken by the data packets to reach from source to destination over a given physical or logical communication channel is termed as average delay. Average Delay of the proposed authentication approach is less as compared to the exisitng authentication approach mentioned in [1,3,8] as shown in Figure 6

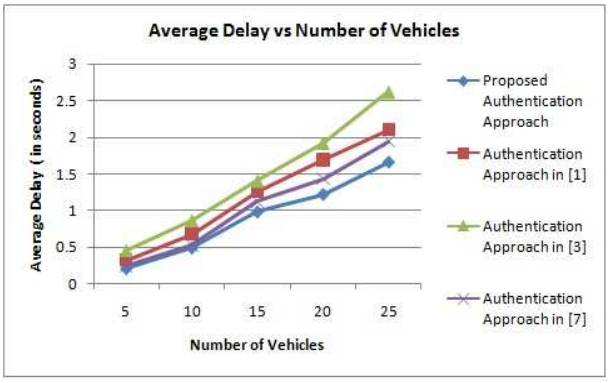


Figure 6: Average delay

7 Conclusions

The discrete event based threat driven authentication approach has been described in this paper. This authentication approach utilizes asymmetric cryptography, reencrypt key and time based arbitrary numbers to provide authentication among vehicles and between vehicles and RSUs. The proposed authentication approach is analyzed by using Petri Nets and Veins framework. With the help of Petri nets model and its reachability graph, it has been observed that the proposed authentication approach acquires the reachability and liveness property. With the help of Veins framework, it has been observed that the proposed authentication approach is better as compared to existing authentication approaches described in [1,3,8] in terms of computational overhead, throughput, packet delivery ratio and average delay. This approach also provides privacy and security among vehicles and between vehicles and RSUs from different types of authentication attacks in VANET.

References

[1] M. Ashritha and C. Sridhar, "Rsu based efficient vehicle authentication mechanism for VANETs," in *IEEE 9th International Conference on Intelligent Systems and Control (ISCO'15)*, pp. 1–5, 2015.

[2] N. B. Bhavesh, S. Maity, and R. C. Hansdah, "A protocol for authentication with multiple levels of anonymity (amla) in VANETs," in *27th International Conference on Advanced Information Networking and Applications Workshops (WAINA'13)*, pp. 462–469, 2013.

[3] M. C. Chuang and J. F. Lee, "Ppas: A privacy preservation authentication scheme for vehicle-to-infrastructure communication networks," in *International Conference on Consumer Electronics, Communications and Networks (CECNet'11)*, pp. 1509–1512, 2011.

[4] S. DasGupta, R. Chaki, and S. Choudhury, "Truval: Trusted vehicle authentication logic for VANET," in *Advances in Computing, Communication, and Control*, pp. 309–322, 2013.

[5] Y. Han, D. Fang, Z. Yue, and J. Zhang, "Schap: The aggregate signcryption based hybrid authentication protocol for VANET," in *International Conference on Internet of Vehicles*, pp. 218–226, 2014.

[6] S. Ibrahim, M. Hamdy, and E. Shaaban, "Towards an optimum authentication service allocation and availability in VANETs," *International Journal of Network Security*, vol. 19, no. 6, pp. 955-965, 2017.

[7] Z. Jianhong, X. Min, and L. Liying, "On the security of a secure batch verification with group testing for VANET," *International Journal of Network Security*, vol. 16, no. 5, pp. 351–358, 2014.

[8] Y. Kim and J. Lee, "A secure analysis of vehicular authentication security scheme of rsus in VANET," *Journal of Computer Virology and Hacking Techniques*, vol. 12, no. 3, pp. 145–150, 2016.

[9] C. T. Li, M. S. Hwang, Y. P. Chu, "A secure and efficient communication scheme with authenticated key establishment and privacy preserving for vehicular ad hoc networks", *Computer Communications*, vol. 31, no. 12, pp. 2803-2814, July 2008.

[10] J. S. Li and K. H. Liu, "A lightweight identity authentication protocol for vehicular networks," *Telecommunication systems*, vol. 53, no. 4, pp. 425–438, 2013.

[11] R. Pradweap and R. Hansdah, "A novel rsu-aided hybrid architecture for anonymous authentication (rahaa) in VANET," in *International Conference on Information Systems Security*, pp. 314–328, 2013.

[12] J. Shao, X. Lin, R. Lu, and C. Zuo, "A threshold anonymous authentication protocol for VANETs," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 3, pp. 1711–1720, 2016.

[13] Y. Wang, H. Zhong, Y. Xu, and J. Cui, "Ecpb: Efficient conditional privacy-preserving authentication scheme supporting batch verification for VANETs." *International Journal of Network Security*, vol. 18, no. 2, pp. 374–382, 2016.

[14] H. T. Wu and W. S. Hsieh, "Rsu-based message authentication for vehicular ad-hoc networks," *Multimedia Tools and Applications*, vol. 66, no. 2, pp. 215–227, 2013.

[15] C. Zhang, X. Lin, R. Lu, P. H. Ho, and X. Shen, "An efficient message authentication scheme for vehicular communications," *IEEE Transactions on Vehicular Technology*, vol. 57, no. 6, pp. 3357–3368, 2008.

[16] C. Zhang, R. Lu, P. H. Ho, and A. Chen, "A location privacy preserving authentication scheme in vehicular networks," in *Wireless Communications and Networking Conference (WCNC'08)*, pp. 2543–2548, 2008.

[17] S. Zeadally, R. Hunt, Y. S. Chen, A. Irwin, and A. Hassan, "Vehicular ad hoc networks (VANETs): status, results, and challenges," *Telecommunication Systems*, vol. 50, no. 4, pp. 217–241, 2012.

[18] S. Zeng, Y. Huang, and X. Liu, "Privacy-preserving communication for VANETs with conditionally anonymous ring signature," *International Journal of Network Security*, vol. 17, no. 2, pp. 135–141, 2015.

Biography

Arun Malik received his Bachelor of Technology from Kurukshetra University in 2008 and Master of Technology from Maharishi Markandeshwar University in 2011.He is pursuing his Ph.D from School of Computer Science and Engineering at Lovely Professional University, Punjab. He has 5 years of teaching experience and published more than 12 papers in journals and conference proceedings.His research interests include wireless networks, Ad hoc Networks, VANET and MANET.

Dr.Babita Pandey is currently with the School of Com-

puter Application at Lovely Professional University, Punjab, India. She obtained her PhD from Indian Institute of Technology-Banaras Hindu University, India in 2009. She has 8 years of teaching experience. She has published more than 80 papers in journals and conference proceedings, some of them are in journal such as such Expert Systems with Applications (Elsevier), Computers in Biology and Medicine (Elsevier), Education and Information Technologies (Springer), SpringerPlus, Nuclear Technology(American Nuclear Society) etc. She is in the editorial board of several journals. Her research interest are intelligent methods in bioinformatics, biomedical signals etc.

An Image Retrieval Algorithm Based on GIST and SIFT Features

Bei Xie, Jiaohua Qin, Xuyu Xiang, Hao Li and Lili Pan

(Corresponding author:Jiaohua Qin)

College of Computer and Information Engineering, Central South University of Forestry and Technology
No. 498, Shaoshan South Road, Changsha 410004, Hunan, P.R. China
(Email: qinjiaohua@163.com)

(Received Feb. 1, 2017; revised and submitted May 14, 2017)

Abstract

Aiming at solving the problem that the single feature cannot represent an image completely, an image retrieval algorithm which combines the global feature and the local feature is proposed. First, the GIST features of the query image and all the images in the image library are extracted. The global similarity of two images is measured by the Euclidean distance. The image database is retrieved by the GIST feature of the query image, and the results are arranged in ascending order of the similarity value. Second, the SIFT features of the query image are extracted as well as the k sub-images which are at the front of the returned results. Then, we perform the feature matching by using BBF algorithm. At last, the retrieval results are returned by sorting the number of matching points in descending order. The experiment is carried out on the improved Caltech101 dataset. Compared with the existing image retrieval algorithms, the proposed image retrieval algorithm not only improves the retrieval accuracy, but also achieves better retrieval efficiency.

Keywords: Feature Extraction; GIST Feature; Image Matching; Image Retrieval; SIFT Feature

1 Introduction

With the rapid development of image sensor and Internet technology, the types and quantity of image data are increasing day by day. How to find the images which can satisfy users' needs from the massive picture library quickly and accurately has become the hotspot of current research. Early image retrieval mainly depends on keywords or text image description. Text matching is used in text-based querying. The text-based image retrieval method requires the image to be annotated beforehand. The artificial annotation is subjective, inadequate, time-consuming and laborious, which cannot meet the demand of retrieval.

Content-based Image Retrieval (CBIR) extracts the feature vector from the underlying features such as color, texture, shape and position of the image [10, 16]. It selects the image set which is closest to the feature vector of the query image as the retrieval result in the image library. With the explosive growth of image database size and the rise of support vector machine (SVM) [19] and AdaBoost [21], machine learning has been widely applied in image retrieval. With the wide spread of privacy protection and information security awareness [11], image encryption [6, 14], encrypted image retrieval [22], image steganography [4, 12, 20] and data hiding [3, 5] are also a new research topics. However, for image retrieval or encrypted image retrieval, the key point is the extraction of the characteristics which can represent the content of the image effectively.

With the deepening of the CBIR research work, more and more image retrieval algorithms based on different content features are proposed. Oliva and Torralba [24] regarded an image as a whole to be detected by the pre-designed feature operator. The calculated multidimensional features recorded the classification information, and the GIST feature was proposed to be a very effective retrieval operator. Though GIST features had many advantages, there were also some shortcomings, such as the information loss in sparse grid computing. Combining with the fuzzy mathematics, Vedran and Ljubovic [13] proposed a FCH (Fuzzy Color Histogram) algorithm in the process of color feature extraction. Compared with the traditional algorithms based on color feature, the results of FCH algorithm combined with fuzzy mathematics are more effective. Ruixia Wang and Guohua Peng [18] proposed a new image retrieval algorithm based on Hesse sparse coding to overcome the shortage that the image spatial structure discards the bag of visual word (BOVW) model. Firstly, the n-words model was established to obtain the local feature representation of the image. Secondly, the second-order Hesse energy function was incorporated into the objective function of standard sparse coding to obtain the Hesse sparse coding formula. Fi-

nally, the n-words sequence was obtained as the encoding feature. The optimal Hesse Coefficient was calculated by Feature Symbol Search Algorithm to get the similarity and return the search results. Lowe proposed the Scale-Invariant Feature Transform (SIFT) [9] method. It used the Difference of Gaussian operator to find the points of interest and described the feature points by the main direction histogram. SIFT features of the image maintained invariant in image rotation, scaling zoom, brightness changes and the angle of view changes. They remain stable in resisting the interference of affine transformation and noise. However, they still had the shortcomings of lower real-time and the features of the target which had smooth edge cannot be accurately extracted. A new improved SIFT-based image retrieval method [15] was proposed for accuracy and time-consuming in SIFT feature points matching. Firstly, the dimensions of image features were reduced by fuzzy C-means clustering method based on weighting of spatial features. Then, the dimension-reduced features were predicted multi-class by the K-D tree algorithm and return the result. Bayes *et al.* [1] proposed the speeded up robust features (SURF) of SIFT. They used the box in different size to do convolution with images so as to approximate Gaussian convolution in SIFT. CC Chen and SL Hsieh [2] *et al.* characterized the SIFT feature as a binary number, which greatly reduced the computation time of the SIFT feature similarity. It improved the retrieval efficiency by the hash method according to the binary feature.

However, it is difficult for the single global feature to represent the details of images which might have the same structure or characteristics. Although the local feature can describe the local details well to get satisfied results, there are still some problems. For example, the operation has a slow speed, there is large correlation between the number of feature points and regions of interest, the number of feature points is quite different in different images. In this paper, an image retrieval algorithm which compromises GIST and SIFT is proposed based on local feature extraction and global feature extraction. The proposed algorithm shows a good retrieval performance.

2 Fusion Feature Extraction

2.1 GIST Feature Extraction

The GIST feature represents the scene information of the image well [23]. Before extracting the GIST feature, the image is divided into several blocks. The blocks are processed by Gabor filters of different scales and different directions in advance, and then average the calculated results of different regions to obtain the required features information.

In order to avoid the loss of information and improve the feature accuracy, the image is divided into several blocks in advance. Do convolution operation on different parts in order and then integrate the results to obtain global GIST feature of the image. Suppose that the original image to be processed is with the size of $M \times N$.Firstly, divide it into $n_b \times n_b$ blocks, and each block represents a region. $n_g = n_b \times n_b$ is used to record the total number of blocks. The different blocks of the image are labeled, denoted by B_b, where $i = 1, \ldots, n_g$. In order to facilitate the calculation and processing, each block is in the same size of $M' \times N'$.

With the self-similarity of Gabor filter, different Gabor filters can be obtained by a number of mathematical transformation and operations when the mother wavelet filter is given. The mother wavelet of Gabor filter is as follows:

$$g(x,y) = \frac{1}{2\pi\sigma_x\sigma_y}\exp[-(\frac{x^2}{\sigma_x^2}+\frac{y^2}{\sigma_y^2})]\times\cos(2\pi f_0 x+\varphi) \quad (1)$$

x, y denotes the position information of the pixel. σ_x, σ_y denotes the Gaussian standard deviation of the x-axis and the y-axis, f_0 denotes the center frequency, and φ denotes the phase shift.

Transform the mother wavelet mathematically and then a set of Gabor filter in different scales and directions can be obtained, the specific formula is as follows:

$$\begin{cases} g_{mn}(x,y) = a^{-m}g(x',y'), a>1 \\ x' = a^{-m}(x\cos\theta + y\sin\theta) \\ y' = a^{-m}(-x\sin\theta + y\cos\theta) \\ \theta = \frac{n\pi}{n+1} \end{cases} \quad (2)$$

Where a^{-m} denotes the scale factor, θ denotes the rotation angle, m denotes the number of scales, and n denotes the number of directions.

There are $m \times n$ Gabor filters after the calculation. Firstly, the same processing is performed on the different regions in the original image, and then the cascade operation is adopted. The result is the Gist feature of the image, as follows:

$$G_i^B(x,y) = cat(I(x,y) * g_{mn}(x,y)), (x,y) \in B_i \quad (3)$$

where G^B denotes the block GIST feature, its dimension is $m \times n \times M' \times N'$, $cat()$ denotes the cascade operation, and $*$ denotes the convolution operation.

For each different filter, the obtained block GIST features are averaged. The results are integrated by rows to get the global GIST feature of the image:$G^G = \{\overline{G_1^B}, \overline{G_2^B}, \ldots, \overline{G_{n_g}^B}\}$, where G^G denotes the global GIST feature, $\overline{G_i^B}$ denotes the mean of the block GIST feature. $\overline{G_i^B}$ is calculated as follows:

$$\overline{G_i^B} = \frac{1}{M' \times N'}\sum_{(x,y)\in B_i} G_i^B(x,y) \quad (4)$$

In this paper, G^G is extracted as the global feature of image, its dimension is $m \times n \times n_g$. G^G can only describe the whole image, but can not express the details of the image well. SIFT, as a local feature description operator, can express the local information of the image by the feature points well.

2.2 SIFT Feature Extraction

The essence of the SIFT algorithm is to find the extreme points and extract their direction, size and position invariants in the scale space. The specific extraction process is divided into the following four steps:

Step 1. Establishment of Scale Space. Firstly, a Gaussian pyramid is constructed by Gaussian smoothing in a pair of image $I(x, y)$. $I(x, y)$ is in the size of $M \times N$, and the $s + 3$ layers of Gaussian images are established in the first-order scale space. The Gaussian convolution kernel is defined as follows:

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-\frac{(x^2+y^2)}{2\sigma^2}} \tag{5}$$

where x, y denotes the position information of the pixel, σ denotes the scale factor.

The bottom Gaussian image is:

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y) \tag{6}$$

From the bottom of the image, the scale factor σ grows in scale proportion k, where $k = 2^{\frac{1}{s}}$, s is 3. When the fourth Gaussian image is generated, the obtained $L(x, y, 2\sigma)$ will be sub-sampled to generate the first image of the second-order Gaussian scale space which is in the size of $\frac{1}{2}M \times \frac{1}{2}N$. The $s + 2$ Difference of Gaussian (DoG) images are obtained by the subtracting of the $s + 3$ sub-images of Gaussian scale space. The formula is as follows:

$$D(x, y, \sigma) = [G(x, y, k\sigma) - G(x, y, \sigma)] * I(x, y) \tag{7}$$

Then the scale sequence of the first-order Gaussian image is: σ, $k * \sigma$, $k^2 * \sigma$, $k^3 * \sigma$, $k^4 * \sigma$, $k^5 * \sigma$. The scale sequence in the corresponding DoG space is:σ, $k\sigma$, $k^2 * \sigma$, $k^3 * \sigma$, $k^4 * \sigma$. The middle three layers are chosen as the range of obtaining the extreme value: $k\sigma$, $k^2 * \sigma$, $k^3 * \sigma$. Deal with the second order in the same way to get the scale sequence of the second-order DoG space: $2k\sigma$, $2k^2 * \sigma$, $2k^3 * \sigma$.

At this time, the last layer of the first-order DoG space is continuous with the first layer of the second order. Pyramid order number is determined by the size of the original image and the top image of pyramid, the formula is as follows:

$$Ordernumber = log_2\{\min(M, N) - t\}, t \in log_2\{\min(M, N)\} \tag{8}$$

where t is the logarithm of the minimum dimension of the top image.

Step 2. Determination of key points. Compare the value of each pixel in the DoG-scale spatial image with the value of its adjacent 26 pixels (9 pixels on the upper layer +8 pixels on the same layer and +9 pixels on the lower layer). If the value of one pixel (x, y) is the local extreme value, then the point is to join the candidate SIFT key points set.

By fitting the three-dimensional quadratic equation, low-contrast points can be found to be eliminated. Do the Taylor expansion for $D(x, y, \sigma)$ at the local extreme point (x_0, y_0, σ_0) to get the quadratic term:

$$D(X) = D + \frac{\partial D^T}{\partial X} X + \frac{1}{2} X^T \frac{\partial^2 D}{\partial X^2} X \tag{9}$$

where $X = (x, y, \sigma)^T$, which represents the offset of the sample points. The extreme point $\hat{X}$ can be obtained by the derivative of the above equation:

$$\hat{X} = -\frac{\partial^2 D^{-1}}{\partial X^2} \frac{\partial D}{\partial X} \tag{10}$$

put Formula (10) into Equation (9):

$$D(\hat{X}) = D + \frac{1}{2} \frac{\partial D^T}{\partial X} \hat{X} \tag{11}$$

If $| D(\hat{X}) | < 0.03$, the point of low contrast is removed. In actual operation, the value of $\frac{\partial D}{\partial X}$ is estimated by the difference of the adjacent pixels of the sampling point. Find the edge point by calculating the Hessian matrix and remove them. The Hessian matrix is as follows:

$$H = \begin{bmatrix} D_{xx} & D_{xy} \\ D_{yx} & D_{yy} \end{bmatrix} \tag{12}$$

where D_{xx}, D_{xy}, D_{yx}, D_{yy} can be estimated by the adjacent difference of sampling points.

Suppose that α and β denote eigenvalues of H, which record the gradient indirection x and y respectively. The traces and determinants of Hessian matrix can be calculated as follows:

$$Tr(H) = D_{xx} + D_{yy} = \alpha + \beta \tag{13}$$

$$Det(H) = D_{xx} D_{yy} - D_{xy}^2 = \alpha\beta \tag{14}$$

Assume that α is the bigger eigenvalue of H and β is the smaller one, let $\alpha = r\beta$, then:

$$\frac{Tr(H)^2}{Det(H)} = \frac{(\alpha + \beta)^2}{\alpha\beta} = \frac{(r\beta + \beta)^2}{r\beta^2} = \frac{(r + 1)^2}{r} \tag{15}$$

If $r = 1$, then $\frac{(r+1)^2}{r}$ gets to a minimum, the value of $\frac{(r+1)^2}{r}$ increases with the increase of r. Larger r value means the more difference between α and β. That is to say, one gradient value is big and the other one is small, which means the information of the edge. Threshold method is adopted to judge the threshold value. If a point satisfies the formula $\frac{Tr(H)^2}{Det(H)} > \frac{(r+1)^2}{r}$, this point is eliminated and $r = 10$ in this paper.

Step 3. Determination of direction parameters of the key points. Calculate the gradient magnitude and direction of all the sampling points which are in the neighborhood window of each key point. The calculation formula is as follows:

$$Grad(x, y) = \sqrt{(L(x + 1, y) - L(x - 1, y))^2 + (L(x, y + 1) - L(x, y - 1))^2} \tag{16}$$

$$\varphi(x,y)=\tan^{-1}\left(\frac{L(x+1,y)-L(x-1,y)}{L(x,y+1)-L(x,y-1)}\right) \quad (17)$$

where $Grad(x,y)$ denotes the gradient magnitude, and $\varphi(x,y)$ denotes the gradient direction.

Count the gradient histogram of 36 directions, and the direction of histogram peak represents the main direction of the key point. The gradient direction information contribution of the sampling points to the center key point is weighted by a Gaussian distribution function. This function is multiplied by the gradient of the sampling point to represent the weight. Regard the direction whose value is more than 80% of the peak in the histogram as the secondary direction of the key point, and then the key point are copied to be directed in different directions. At this time, the scale and specific orientation of a key point are found.

Step 4. Key point feature description. Rotate the axis to the direction of the key pixel, and then take the key pixel as the center to get a 16×16 window. The window is divided into 4×4 sub-window. Use Gaussian blur method to increase the weight of the neighborhood points which are closer to the key point, and reduce the weight values of those distant points. Calculate the gradient accumulation value of 8 directions in each sub-region to obtain $4\times4\times8=128$ dimensional feature descriptor.

3 Feature Fusion Based on Local and Global Features

3.1 Feature Matching Algorithm Based on BBF

Each feature description algorithm focuses on the different side of an image. The similarity measure of the local feature operator and the global feature operator are also not the same. As the vector of global feature is a single vector, its similarity measure is simpler than local feature. The similarity can be calculated directly by similarity distance formula. In this paper, Euclidean distance [25] is used as the similarity measure function:

$$R=\parallel G_1-G_2\parallel \quad (18)$$

where R denotes the measure of similarity, G_1 and G_2 denote global eigenvectors of different images, and $\|\cdot\|$ denotes the 2-norm of the computed vector.

Local feature point matching is to find the most similar points in the two images. Take a key point in one image and find the two key points which have the smallest similarity distance with the former. If the ratio between the nearest distance and the second nearest distance is less than a certain threshold, this pair will be accepted. Let $ratio=\frac{d1}{d2}$, where $d1$ is the nearest distance and $d2$ is the second nearest distance. Define the matching function T (ratio) as follows:

Algorithm 1: Feature point matching algorithm

Input: Eigenvector set $DES1$, $DES2$

Out put: Successful matched pairs

Initialization $des1\leftarrow\emptyset, des2\leftarrow\emptyset, pair=0$
Use KD-Tree to index all the elements of $DES2$
for i=1, $\cdots$,m **do**
  $DES1=[U_1;U_2;\cdots;U_m], DES2=[V_1;V_2;\cdots;V_n]$
  According to BBF algorithm, obtain the approximate 2-nearest neighbors V_j, V_k of U_i which is from $DES1$ in KD-Tree
  if $T(\frac{d(U_i,V_j)}{d(U_i,V_k)})=1$ **then** $des1\leftarrow U_i, des2\leftarrow V_j, pair=pair+1$
  else Remove U_i (no valid matched point)
end for

Algorithm 2: Extraction algorithm of global GIST feature

Input: Image $I(x,y)$ with the size of $M\times N$

Out put: Feature vector G^G

for j=1, $\cdots$,m **do**
  for k=1, $\cdots$, n **do**
  Calculate $m\times n$ Gabor filters $g_{jk}(x,y)$ by Formula (2)
  end for
end for
Divide $I(x,y)$ into $n_g=n_b\times n_b$ blocks and label those different blocks with B_i
for i=1, $\cdots$, n_g **do**
  Calculate $G_i^B(x,y)$ by Formula (3)
  Calculate $\overline{G_i^B}$ by Formula (4)
end for
$G^G=\{\overline{G_1^B},\overline{G_2^B},\ldots,\overline{G_{n_g}^B}\}$

$$T(ratio)=\begin{cases}1 & if ratio<\varepsilon, Accepted\\ 0 & else, Not\ accepted\end{cases} \quad (19)$$

For each local feature eigenvector in the query image, it needs to compare similarity distance with all eigenvectors in other image. In this paper, Euclidean distance is used as the distance function. Obviously, exhaustive search can achieve precise positioning, but it is very inefficient in the actual application. In this paper, a BBF [8] search algorithm is chosen to make a trade-off between accuracy and efficiency. Specific feature point matching algorithm is as Algorithm 1.

3.2 Global Feature and Local Feature Extraction Algorithm

The extraction of GIST feature treats the entire image as a whole to perform feature detection through a predesigned feature operator and records the relevant category information with the calculated multidimensional feature. There is no need to calculate a lot of complex minutiae in the whole process. Therefore, the interference from some small noise to the classification can be avoided and the additional error caused by unnecessary processing can be reduced. So the feature has a good advantage in the initial division of data. The detailed process of feature extraction is shown in Algorithm 2:

Though GIST feature have many advantages, there are also some corresponding shortcomings, such as the loss of information caused by sparse grid computing and poor description of the local information in an image. As a local feature, the SIFT feature has strong robustness to image rotation, scale scaling, brightness variation, angle change, affine transformation and noise, but there are still some

Algorithm 3: Extraction algorithm of local SIFT feature

Input: Image $I(x, y)$ with the size of $M \times N$
Out put: Eigenvector set des

Initialization $des \leftarrow \emptyset$
Calculate the Gaussian pyramid order n by the Formula (8)
for i=1, $\cdots$, n **do**
  for j=1, $\cdots$,m **do**
  Calculate the $m+1$ Gaussian images with the scale factor coefficient of k^j according to Formula (6)
  end for
  Sub-sample the $m-1$ Gaussian images as a starting image of the next order
  Calculate $D = \{L_{k+1} - L_k\}_{k=2}^{m-1}$ by Formula (7)
  Get extreme points $D(x_0, y_0, \sigma_0)$ of $D = \{L_{k+1} - L_k\}_{k=2}^{m-1}$ and remove the edge points by Formula (9)-(15)
  Calculate $Grad$ and φ of a certain neighborhood pixels for the key points by Formulas (16) and (17) and Gaussian weighted gradient $\{Grad_i\}_{i=1}^{36}$ in 36 directions,if $Grad_k = \max(\{Grad_i\}_{i=1}^{36})$, the direction of key point is the direction of $Grad_k$
  Generate des_i by Step (4), $des \leftarrow des \bigcup des_i$
end for

Algorithm 4: Fusion algorithm of GIST and SIFT

Input: Query image s, image library $\Phi = \{x_i\}_{i=1}^{n}$
Out put: Query result image set $\Psi = \{y_i\}_{i=1}^{m}$

Initialization $G \leftarrow \emptyset, R \leftarrow \emptyset, T \leftarrow \emptyset$
for i=1, $\cdots$, n **do**
  Set x_i as the input to run Algorithm 2, the Gist feature vector G_i of x_i is obtained, $G \leftarrow G \bigcup G_i$
end for
Set s as the input to run Algorithm 2, obtain Gist feature vector g of s
for i=1, $\cdots$, n **do**
  Calculate $R_i = \| g - G_i \|$ by Formula (18), $R \leftarrow R \bigcup R_i$
end for
Order R in ascending to obtain the $\{R_i\}_{i=1}^{n}$ $(R_1 < R_2 < \cdots < R_n)$, return corresponding image set $\{x_i\}_{i=1}^{k}$ of $\{R_i\}_{i=1}^{k}$
Set s as the input to run Algorithm 3, obtain the SIFT feature vector set S
for i=1, $\cdots$,k **do**
  Set x_i as the input to run Algorithm 3, obtain the SIFT feature vector set S_i
  Set S, S_i as input to run Algorithm 1, calculate the number T of matched vectors, $T \leftarrow T \bigcup T_i$
end for
Order T in descending to obtain $\{T_i\}_{i=1}^{k}(T_1 > T_2 > \cdots > T_k)$, return corresponding image set $\Psi = \{y_i\}_{i=1}^{m}$ of $\{T_i\}_{i=1}^{m}$

problems. For example, it has a low real-time property and it cannot extract feature points accurately to edge smoothing target when the feature points is less. The specific algorithm of SIFT feature is described in Algorithm 3.

3.3 Retrieval Algorithm Based on the Fusion of GIST and SIFT

In order to solve the problem of large amount of image data in the current image retrieval technology, this paper adopts a primary classification method to improve the retrieval efficiency. When the human eye identify the images, a global recognition is first cared and then we will care about the local details, namely, we first consider whether the overall sensory of two images are similar, and then consider whether the local features are more similar. So a fusion algorithm design process is proposed in Figure 1. In this paper, the GIST and SIFT features are fused according to the idea of primary classification and the idea of the global recognition first.

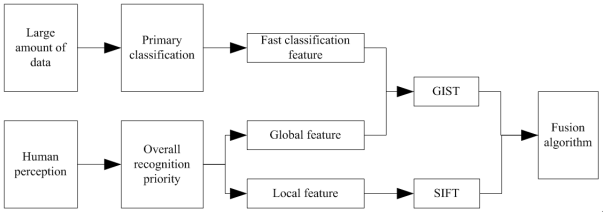


Figure 1: Conception process of fusion algorithm

Each single feature description algorithm focuses on different side when expressing the image content. So when using a single feature description to do image retrieval, the results will be at a loss. GIST features only need very little characteristic dimension to describe the scene of an image, which is very effective in image classification and retrieval and conducive to the rapid retrieval of images. However, as a global feature, the GIST also has the common problem of global characteristics, lack of ability to distinguish details. SIFT as a local feature can only make up for the shortcomings of GIST features. The GIST and SIFT features are complementary to each other in a certain extent. Therefore, this paper designs an image retrieval algorithm described in Algorithm 4 based on the fusion of GIST and SIFT.

4 Evaluation Criteria for Image Retrieval

In this paper, we use recall R, precision P, $F_1 - measure$ [17] and average accuracy MAP to judge the system.

$$P = \frac{A}{(A+B)}$$
$$R = \frac{A}{(A+C)}$$

where A denotes the number of related images, $A+B$ denotes the number of result images, $A+C$ is the total number of related images.

$F_1 - measure$ is defined as:

$$F_1 - measure = \frac{(\beta^2+1)PR}{(\beta^2 P + R)}$$

When $\beta = 1$, $F_1 = \frac{2PR}{P+R}$ is the $F_1 - measure$.

The graphic representation is shown in Figure 2.

The average accuracy (MAP) is calculated as follows:

$$MAP(Q) = \frac{1}{\mid Q \mid} \sum_{j=1}^{|Q|} \frac{1}{m_j} \sum_{k=1}^{m_j} P(R_{jk}) \qquad (20)$$

where, $P(R_{jk}) = \frac{k}{rank(k)}$, $AP(j) = \frac{1}{m_j} \sum_{k=1}^{m_j} P(R_{jk})$

$AP(j)$ is the average precision of the query image j, Q denotes the set of all query images, $\mid Q \mid$ denotes the number of query images, m_j is the number of images related to the query image Q_j in the image database, $rank(k)$ denotes the rank of related image in the returned result.

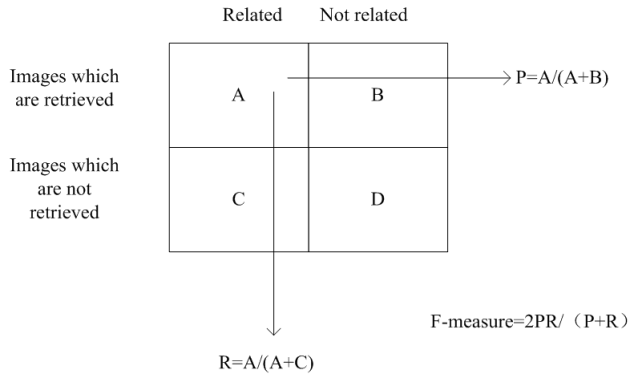


Figure 2: Precision and Recall relationship diagram

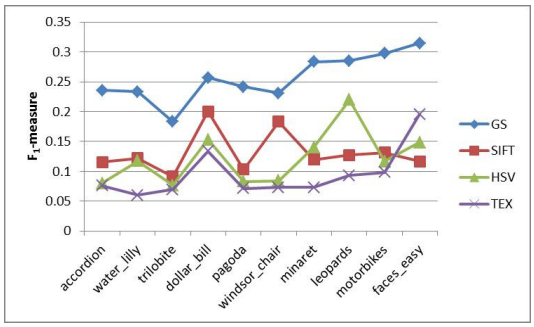


Figure 4: The $F_1-measure$ contrastive figure of different algorithms on Caltech101 testing dataset

5 Experimental Results and Analysis

5.1 Testing Image Library

The Caltech101 dataset [7] contains 101 scene types and 1 background type. There are 9144 images in whole, including animals, vehicles, flowers, people, soccer and so on. They are significant different in shape. The number of each type is from 31 to 800. In this paper, we choose 10 categories from the Caltech101 data set as follows: accordion, water-lilly, trilobite, dollar-bill, pagoda, Windsor-chair, minaret, Leopards, Motorbikes and Faces-easy. The Faces-easy class is composed of different human faces. Twenty-five images are randomly selected from the first nine categories. Do blurring, affine transformation and brightness change operation to the chosen images. Choose 4 different images of 25 people in the Faces-easy class randomly. These 10 types of images will form a Caltech 101 testing dataset which contains 1000 images. Figure 3 shows two images of each type from the testing dataset.

Figure 3: Part of the Caltech 101 testing dataset

5.2 Experimental Results

The experiment is carried out on the testing dataset. In Algorithm 1, the related parameter is set as $\varepsilon = 0.6$. In Algorithm 2, the relevant parameters are set as $m = 4$ and $n = 8$. In Algorithm 4, the relevant parameter is set as $k = 100$. Figure 4 is the comparison about the comprehensive evaluation index $F_1 - measure$ between the algorithm in this paper (GS), HSV color histogram retrieval algorithm (HSV), SIFT feature retrieval algorithm (SIFT) and texture feature retrieval algorithm (TEX) on Caltech101 testing dataset. In the figure, the abscissa represents the classification of image and the ordinate is $F_1 - measure$. Figure 5 is an example of the results of this algorithm in the Caltech 101 testing dataset, and the first 20 results are returned.

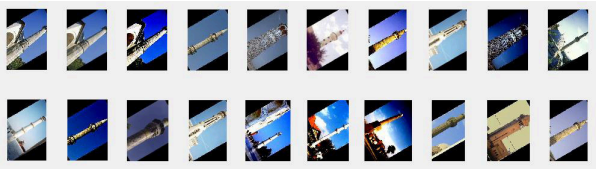


Figure 5: An example of retrieval results on Caltech101 testing dataset

It can be seen from Figure 4 that the retrieval accuracy is improved to a great extent compared with the SIFT-based retrieval algorithm, HSV color histogram and texture-based algorithm.

In image retrieval, people cares more about whether the main body of two pictures are the same thing. Therefore, when the MAP value is calculated, the original image and the three kinds of transformed images are used as correlation images. Figure 6 shows the comparison of the average retrieval accuracy of the GS algorithm, the GIST algorithm and the SIFT algorithm.

The Figure 6 shows that the retrieval algorithm proposed in this paper has more advantages than the GIST retrieval algorithm in average accuracy, but it is worse

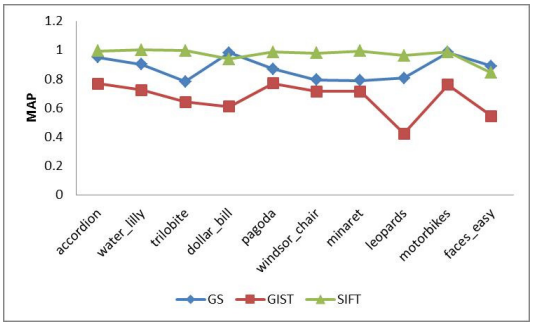


Figure 6: The MAP contrastive results of different algorithms on Caltech101 testing dataset

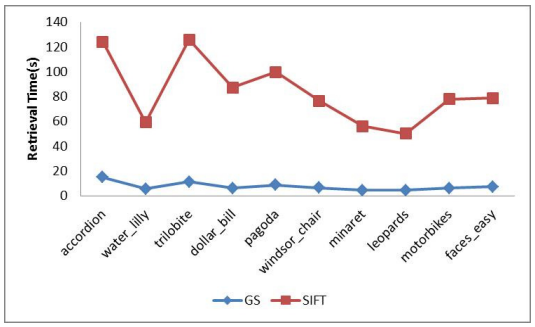


Figure 7: The Time contrastive results of different algorithms on Caltech101 testing dataset

than the SIFT algorithm. However, the real-time retrieval performance in image retrieval is also a very important index to measure retrieval performance. Figure 7 is the retrieval time performance comparison between the proposed retrieval algorithm and the sift algorithm. It can be seen that the retrieval time of the SIFT algorithm is far more than the proposed algorithm. So the proposed retrieval algorithm not only ensures the retrieval accuracy but also satisfies the real-time requirement in the real retrieval.

The above results prove that the retrieval performance of the proposed retrieval algorithm in this paper is good. It not only performs well in those images which have the same or local similar scene but also keeps robustness to the fuzzy, affine and brightness change. The better real-time also conform to the real application environment.

6 Conclusion

This paper proposed an image retrieval algorithm which combined global and local features. Firstly, the GIST features of all the images in the image database were extracted, and then the k nearest neighbors of the query image in the image database were returned according to Euclidean distance. Secondly, we extracted the SIFT feature of the k nearest neighbors results as well as the query image, and performed points matching according to the BBF searching algorithm. The results were returned according to the descending order of the matching points number. Finally, a retrieval experiment was carried out on the improved dataset of Caltech 101. The results showed that the new retrieval algorithm not only improved the retrieval precision, but also had good performance in real-time.

Acknowledgments

This work is supported by the Natural Science Foundation of China (Nos. 61772561, 61202496, U1405254), Science Research of Hunan Provincial Education Department (Grant No. 16C1659), A Project Funded by the Priority Academic Program Development of Jiangsu Higher Education Institutions.

References

[1] H. Bay, T. Tuytelaars, L. V. Gool, "Surf: Speeded up robust features," *Proceedings of European Conference on Computer Vision*, vol. 110, no. 3, pp. 404-417, 2006.

[2] C. C. Che, H. S. Lin, "Using binarization and hashing for efficient SIFT matching," *Journal of Visual Communication and Image Representation*, vol. 30, pp. 86-93, 2015.

[3] L. C. Huang, L. Y. Tseng, M. S. Hwang, "A reversible data hiding method by histogram shifting in high quality medical images", *Journal of Systems and Software*, vol. 86, no. 3, pp. 716–727, Mar. 2013.

[4] M. S. Hwang, C. C. Chang, K. F. Hwang, "Digital watermarking of images using neural networks", *Journal of Electronic Imaging*, vol. 9, no. 4, pp. 548–556, Jan. 2000.

[5] B. Jana, "Dual image based reversible data hiding scheme using weighted matrix," *International Journal of Electronics and Information Engineering*, vol. 5, no. 1, pp. 6-19, 2016.

[6] C. Jin, H. Liu, "A color image encryption scheme based on arnold scrambling and quantum chaotic", *International Journal of Network Security*, vol. 19, no. 3, pp. 347-375, 2017.

[7] F. F. Li, F. Rob, P. Pietro, "Learning generative visual models from few training examples: an incremental bayesian approach tested on 101 object categories," *Computer Vision and Image Understanding*, vol. 106, no. 1, pp. 59-70, 2004.

[8] K. L. Li, S. Sun, "Image copy and paste tamper detection based on improved SIFT algorithm," *Computer Science*, vol. 43, no. s1, pp. 179-183, 2016.

[9] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91-110, 2004.

[10] J. H. Qin, Q. Y. Wang, X. Y. Xiang, B. Xie, L. L. Pan, H. J. Huang, "Ceramic tile image retriveval method based on visual feature," *Journal of Computational and Theoretical Nanoscience*, vol. 12, no. 11, pp. 4191-4195, 2015.

[11] J. H. Qin, R. X. Sun, X. Y. Xiang, H. Li, H. J. Huang, "Anti-fake digital watermarking algorithm based on QR codes and DWT", *International Journal of Network Security*, vol. 18, no. 6, pp. 1102-1108, 2016

[12] M. Shobana, "Efficient x-box mapping in stego-image using four-bit concatenation," *International Journal of Electronics and Information Engineering*, vol. 1, no. 1, pp. 29-33, 2014.

[13] H. Supic, V. Ljubovic, "A compact color descriptor for image retrieval," in *IEEE International Symposium on Information, Communication and Automation Technologies*, pp. 1-5, 2013.

[14] O. Wahballa, A. Wahaballa, F. G. Li, I. I. Idris and C. X. Xu, "Medical image encryption scheme based on arnold transformation and ID-AK protocol," *International Journal of Network Security*, vol. 19, no. 5, pp. 776-784, 2017.
[15] F. Wang, "An improved method of image retrieval based on SIFT," *Digital Technology and Application*, no. 1, pp. 139-141, 2016.
[16] Q. Y. Wang, J. H. Qin, X. Y. Xiang, B. Xie, H. J. Huang, L. L. Pan, "Agricultural product trademark image retrieval method," *Journal of Computational and Theoretical Nanoscience*, vol. 12, no. 11, pp. 4010-4016, 2015.
[17] R. X. Wang, G. H. Peng, "An image retrieval method with sparse coding based on riemannian manifold," *Acta Automatica Sinica*, vol. 43, no. 5, pp. 778-788, 2017.
[18] R. X. Wang, G. H. Peng, "Hesse sparse representation under n-words model for image retrieval," *Journal of Electronics and Information Technology*, vol. 38, no. 5, pp. 1115-1122, 2016.
[19] X. Z. Wen, L. Shao, Y. Xue, *et al*, "A rapid learning algorithm for vehicle classification," *Information Sciences*, vol. 295, no. 1, pp. 395-406, 2015.
[20] C. C. Wu, S. J. Kao, and M. S. Hwang, "A high quality image sharing with steganography and adaptive authentication scheme", *Journal of Systems and Software*, vol. 84, no. 12, pp. 2196–2207, 2011.
[21] Z. H. Xia, X. H. Wang, X. M. Sun, *et al*, "Steganalysis of least significant bit matching using multi-order differences", *Security and Communication Networks*, vol. 7, no. 8, pp. 1283-1291, 2014.
[22] Z. H. Xia, X. H. Wang, L. G. Zhang, *et al*, "A privacy-preserving and copy-deterrence content-based image retrieval scheme in cloud computing," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 11, pp. 2594-2608, 2016.
[23] T. Xu, *An Image Retrieval Method Based on Visual Attention Model and Gist Feature*, Guangxi Normal University, 2014.
[24] Z. Yang, J. Gao, Z. Xie, *et al*, "Scene classification of local Gist feature matching kernel," *Journal of Image and Graphics*, vol. 18, no. 3, pp. 264-270, 2013.
[25] Y. H. Zheng, J. Byeungwoo, D. H. Xu, *et al*, "Image segmentation by generalized hierarchical fuzzy C-means algorithm," *Journal of Intelligent and Fuzzy Systems*, vol. 28, no. 2, pp. 961-973, 2015.

Biography

Bei Xie received his BS in electrical engineering and automation from Wuhan Polytechnic University, China, in 2013. He is currently pursuing his MS in computer application technology at College of Computer Science and Information Technology, Central South University of Forestry and Technology, China. His research interests include information security and image processing.

Jiaohua Qin received her BS in mathematics from Hunan University of Science and Technology, China, in 1996, MS in computer science and technology from National University of Defense Technology, China, in 2001, and PhD in computing science from Hunan University, China, in 2009. She is a professor at College of Computer Science and Information Technology, Central South University of Forestry and Technology, China. Her research interests include network and information security, image processing and pattern recognition.

Xuyu Xiang received his BS in mathematics from Hunan Normal University, China, in 1996, MS degree in computer science and technology from National University of Defense Technology, China, in 2003, and PhD in computing science from Hunan University, China, in 2010. He is a professor at Central South University of Forestry and Technology, China. His research interests include network and information security, image processing, and internet of things.

Hao Li received his BS in computer science and technology from Zhengzhou University, China, in 2015. He is currently pursuing his MS in computer application technology at College of Computer Science and Information Technology, Central South University of Forestry and Technology, China. His research interests include information security and image processing.

Lili Pan received her MS in software engineering from Hunan University,China,in 2004, and PhD in computer application technology from Hunan University,China,in 2009.She is an asscoiate professor at College Science and Information Technology,Central South University of Forestry and Technology, China.Her research interests include image processing and software testing.

Diffie-Hellman Type Key Exchange, ElGamal Like Encryption/Decryption and Proxy Re-encryption Using Circulant Matrices

Chitra Rajarama[1], Jagadeesha Narasimhamurthy Sugatoor[2], and T. Yerri Swamy[3]
(Corresponding author: Chitra Rajarama)
Information Science, Engineering, NIE Institute of Technology[1]
Mysuru-570018, Karnataka, India
Electronics, Communications Engineering, PES Institute of Technology, Management[2]
Shimoga-577204, Karnataka, India
Computer Science, Engineering, KLE Institute of Technology[3]
Hubli - 580030, Karnataka, India
(Email: chitramanuel@yahoo.co.in)
(Received Mar. 22, 2017; revised and accepted Nov. 11, 2017)

Abstract

New methods for Diffie-Hellman type key exchange, ElGamal like encryption decryption and proxy re-encryption, using circulant integer matrices as the private keys, are described. Arithmetic operations are carried out using modular arithmetic to provide secrecy as well as to limit the size of the elements of the key matrices. Here Bidirectional proxy re-encryption is realized using circulant matrices. In the proposed proxy re-encryption technique, we use only matrix multiplication and inversion. Here, the proxy re-encryptors can be easily cascaded. Our scheme is efficient and simple to implement.

Keywords: Circulant Integer Matrices; Diffie-Hellman Type Key Exchange; ElGamal-like Encryption Decryption; Proxy Re-encryption

1 Introduction

The most popular key exchange technique over an unsecure channel is Diffie-Hellman (DH) Key Agreement protocol [5, 11, 21]. In our scheme we use integer matrices as the parameters of the cryptosystem so that the effective size of the keys can be large with smaller sized integers as the elements of the key matrices. The elements of the matrices used belong to the finite field Z_p where in all the numbers are integers in the range 0 to (p–1) and all the arithmetic operations are carried out with respect to modulo p where p is a suitable prime number. In our scheme the private keys used are circulant matrices. ElGamal [6] encryption/decryption is a public key cryptosystem where the cipher text has two components. We use circulant matrices to realize the ElGamal scheme.

Proxy re-encryption [2, 3] basically delegates the decryption process to a third party by re-encrypting the ciphertext. Proxy re-encryption has become an important tool in digital rights management schemes in cloud computing. Our main contribution is the application of circulant matrices in a new way for bidirectional proxy re-encryption and decryption.It involves matrix multiplication and inversion in Z_p.

The paper is organized as follows. Section 2 contains brief information about previous work in this field. Section 3 gives preliminary symbols, notations and definitions. Section 4 describes the Diffie-Hellman key exchange using matrices. In Section 5, ElGamal encryption/decryption is given. Section 6 describes proxy re-encryption using matrices. In Section 7, we discuss matrix keys versus scalar keys in cryptography, vulnerabilities of circulant matrices and a brief comparison with other methods. Conclusion is presented in Section 8.

2 Previous Work

Hill Cipher [9] is the earliest work where a square nonsingular matrix is used as the symmetric key. Matrix multiplication is used for encryption and multiplication by the inverse of that matrix is used for decryption. All arithmetic operations are carried out in the finite field domain. Maximum Distance Separable matrices have been proposed for cryptography by a few authors [8,12]. Use of circulant matrices in cryptography is described and discussed in several research works [10, 15, 17, 18]. Products of commutative matrices as public keys are used for Diffie-Hellman type key exchange [19, 20]. Our work also uses product of matrices but in a different way as described

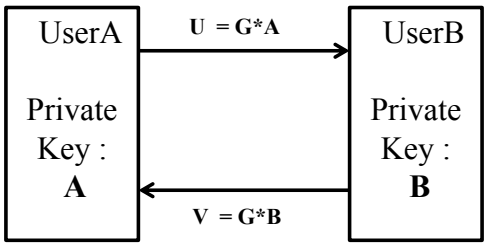


Figure 1: Diffie-Hellman type key exchange

later.

Several authors have based their proxy re-encryption algorithms on bilinear maps [1,2,14,22,23]. But we could not find any earlier work on the use of matrices for proxy re-encryption.

3 Symbols, Notations and Definitions

Consider a two user system as shown in Figure 1. The two users are designated as User A and User B. We assume a bidirectional communication link between User A and User B.

3.1 Circulant Matrix

A circulant matrix [4,13] is a square matrix where, given the first row, the successive rows are obtained by cyclically right shifting the present row by one element. Thus the i^{th} row of a circulant matrix of size (n x n) is obtained by cyclically right shifting the $(i-1)^{th}$ row by one position, for i = 2 to n, given the first row. Let the first row be the row vector, [$c(1)$, $c(2)$, ..., c(n–1), $c(n)$]. Then the circulant matrix **C** is obtained as

$$\mathbf{C} = \begin{bmatrix} c(1) & c(2) & \cdots & c(n) \\ c(n) & c(1) & \cdots & c(n-1) \\ \cdots & \cdots & \cdots & \cdots \\ c(2) & c(3) & \cdots & c(1) \end{bmatrix}$$

The elements of the first row are chosen such that gcd (elements of first row) = 1. This condition assures that the rank of the circulant matrix **C** is n. The most important property of circulant matrices is they are multiplicatively commutative. In our proposed method, we use circulant matrices which belong to the closed linear group GL(n, p) [3]. A Linear group GL(n, p) [16] represents non-singular matrices of size nxn over a finite field (Galois Field) GF(p) or Zp.

3.2 Members of The Cryptosystem

Private keys of User A and User B are **A** and **B** respectively which are circulant matrices of size (nxn). Matrices **A** and **B** belong to GL(n, p). The elements of the first rows of **A** and **B** are chosen so that the rank of both **A** and **B** is n. The generator matrix for this DH system is **G** which is a rectangular matrix of size (n–1) x n. The elements of **G** belongs to Zp. The elements of the generator matrix **G** are so chosen that the rank of **G** is (n–1). That is, rank(**G**) =(n–1). The public key of User A is denoted by matrix **U** and it is generated as

$$\mathbf{U} = \mathbf{G}^*\mathbf{A} \tag{1}$$

The size of **U** is ((n–1) x n) x (n x n) = (n–1) x n. By knowing **U** and **G**, the private key **A** cannot be determined, because the left modular multiplicative inverse of **G** does not exist. In our scheme, **G** and **A** are so chosen that the rank of **U** = **G***A is (n–1). The public key of User B is denoted by matrix **V** and it is given by,

$$\mathbf{V} = \mathbf{G}^*\mathbf{B} \tag{2}$$

The size of **V** is ((n–1) x n) x (n x n) = (n–1) x n. The matrix **B** is so chosen that the rank of **G***B is (n–1). Here also, **B** cannot be determined by knowing **V** and **G**. In our scheme, **U**, **V**, **G** and scalars n, p are in public domain while **A** and **B** are held private. All matrix multiplications are carried out in the finite field Zp.

4 Diffie-Hellman Type Key Exchange

User A sends matrix **U** to User B and User B sends matrix **V** to User A over the unsecured channel. User A calculates the common key $\mathbf{K_A}$ as

$$\mathbf{K_A} = \mathbf{V}^*\mathbf{A} \tag{3}$$

The size of $\mathbf{K_A}$ is ((n–1) x n) x (n x n) =(n–1) x n. Similarly, User B calculates the common key $\mathbf{K_B}$ as

$$\mathbf{K_B} = \mathbf{U}^*\mathbf{B} \tag{4}$$

The size of $\mathbf{K_B}$ is ((n–1) x n) x (n x n) = (n–1) x n. From Equations (2) and (3),

$$\mathbf{K_A} = \mathbf{G}^*\mathbf{B}^*\mathbf{A} \tag{5}$$

From Equations (1) and (4),

$$\mathbf{K_B} = \mathbf{G}^*\mathbf{A}^*\mathbf{B} \tag{6}$$

Since **A** and **B** are circulant matrices of size (n x n), they are multiplicatively commutative [13] as

$$\mathbf{G}^*\mathbf{B}^*\mathbf{A} = \mathbf{G}^*\mathbf{A}^*\mathbf{B} \tag{7}$$

From Equations (5), (6) and (7), the common keys of User A and User B are equal and the system common key **K** is

$$\mathbf{K} = \mathbf{K_A} = \mathbf{K_B} = \mathbf{G}^*\mathbf{A}^*\mathbf{B} = \mathbf{G}^*\mathbf{B}^*\mathbf{A} \tag{8}$$

The size of **K** is (n–1) x n. In Equations (3), (4), (5), (6) and (7), the results of the matrix multiplications are with respect to modulo p, even though the mod operation is not explicitly shown in those equations. Therefore, **K**, $\mathbf{K_A}$ and $\mathbf{K_B}$ also belong to Zp. Matrices **A**, **G** and **B** are so chosen that **K** given by Equation (8) has a rank of (n–1) so that its right modular inverse exists.

Example 1. *Let n = 4. The value of p is taken as 23. Matrices **G**, **A** and **B** are chosen as*

$$\boldsymbol{G} = \begin{bmatrix} 10 & 4 & 11 & 3 \\ 7 & 9 & 11 & 10 \\ 3 & 6 & 8 & 0 \end{bmatrix}$$

$$\boldsymbol{A} = \begin{bmatrix} 10 & 1 & 3 & 3 \\ 3 & 10 & 1 & 3 \\ 3 & 3 & 10 & 1 \\ 1 & 3 & 3 & 10 \end{bmatrix} \quad \boldsymbol{B} = \begin{bmatrix} 3 & 15 & 6 & 3 \\ 3 & 3 & 15 & 6 \\ 6 & 3 & 3 & 15 \\ 15 & 6 & 3 & 3 \end{bmatrix}$$

***U** and **V** are found to be*

$$\boldsymbol{U} = \begin{bmatrix} 10 & 0 & 15 & 14 \\ 2 & 22 & 9 & 21 \\ 3 & 18 & 3 & 2 \end{bmatrix} \quad \boldsymbol{V} = \begin{bmatrix} 15 & 6 & 1 & 21 \\ 11 & 18 & 10 & 17 \\ 6 & 18 & 17 & 4 \end{bmatrix}$$

*Without modulus operation, **V*A** and **U*B** are*

$$\boldsymbol{V*A} = \begin{bmatrix} 192 & 141 & 124 & 274 \\ 211 & 272 & 202 & 267 \\ 169 & 249 & 218 & 129 \end{bmatrix}$$

$$\boldsymbol{V*B} = \begin{bmatrix} 330 & 279 & 147 & 297 \\ 441 & 249 & 432 & 336 \\ 261 & 180 & 333 & 198 \end{bmatrix}$$

*With modulus p, mod(**V*A**, p) = mod(**U*B**, p) = $\boldsymbol{K_A}$ = $\boldsymbol{K_B}$ = **K** and with p=23, we get*

$$\boldsymbol{K} = \boldsymbol{K_A} = \boldsymbol{K_B} = \begin{bmatrix} 8 & 3 & 9 & 21 \\ 4 & 19 & 18 & 14 \\ 8 & 19 & 11 & 14 \end{bmatrix}$$

5 ElGamal Type Encryption and Decryption

ElGamal method [6] is an asymmetric key algorithm for public key cryptography. It is essentially based on Diffie-Hellman key exchange principle. ElGamal method that uses matrix keys is described in this section. All operations are with respect to mod p.

Let **M** be the message matrix whose elements are integers in the range 0 to (p–1). That is, the elements of **M** belong to Zp. The size of **M** is (n–1) x (n–1). User A encrypts **M** and sends it to User B. Matrices **A**, **B**, **G**, **U**, **V** and scalar p are same as described in Section 4. We assume that User A has already received **V** from User B. The encryption by User A is done by generating two crypto terms **U** (same as given by (1)) and **W** as

$$\begin{aligned} \mathbf{U} &= \mathbf{G*A} \\ \mathbf{W} &= \mathbf{M*V*A} \end{aligned} \tag{9}$$

From (2), (8) and (9),

$$\mathbf{W} = \mathbf{M*G*B*A} = \mathbf{M*G*A*B} \tag{10}$$

In the light of (8), Equation (10) can be rewritten as

$$\mathbf{W} = \mathbf{M*K_B} \tag{11}$$

The size of $\mathbf{K_B}$ is (n–1) x n. Since $\mathbf{K_B}$ is not a square matrix, it has no direct inverse. But, (11) can be solved for M using the pseudo inverse of $\mathbf{K_B}$ as

$$\mathbf{M} = \mathbf{W*(K_B)^\dagger} \tag{12}$$

Here, $\mathbf{(K_B)^\dagger}$ is the pseudo right modular inverse of $\mathbf{K_B}$ and is given by,

$$\mathbf{(K_B)^\dagger = K_B^T * (K_B * K_B^T)^{-1}} \tag{13}$$

We choose the crypto-parameters **G**, **A** and **B** such that $\mathbf{K_B}$ is a full rank matrix. Now, User A, the encrypter, sends the pair (**U**, **W**) to User B who is the intended decrypter.

5.1 Decryption at User B

On receiving (**U**, **W**), User B calculates $\mathbf{K_B}$ using (4). Then, he determines $\mathbf{(K_B)^\dagger}$ using (13) and consequently recovers **M** from (12). Results of all operations are calculated with respect to mod p.

5.2 Matrix Inverse in finite field Zp

Consider a square integer matrix **E** of size mxm and rank m, whose elements belong to Zp. We calculate the matrix **L**, the inverse of **E** with respect to mod p using the MatModInv(**E**, p) function [18] as

$$\mathbf{L} = \mathbf{MatModInv}(\mathbf{E}, p) \tag{14}$$

Here, **L** is found such that,

$$\mathbf{mod}(\mathbf{E*L}, p) = \mathbf{mod}(\mathbf{L*E}, p) = \mathbf{Im} = \mathbf{eye}(m) \tag{15}$$

where, $\mathbf{Im} = \mathbf{eye}(m)$ is the Identity Matrix of size mxm. Then **L** is the Matrix Inverse of **E** in the finite field Zp. The elements of **L** are integers in Zp.

Now, consider $\mathbf{W = M*K_B}$ as given by (11). The size of $\mathbf{K_B}$ is (n–1)xn. Since $\mathbf{K_B}$ is not a square matrix, it has no direct inverse. But,(11) is solved for M by post multiplying both sides of (11) by $\mathbf{(K_B)^T}$ which is the transpose of $\mathbf{K_B}$, to get

$$\mathbf{W * K_B^T = M * K_B * K_B^T} \tag{16}$$

The size of $(\mathbf{K_B} * \mathbf{K_B^T})$ is ((n–1)xn) x (nx(n–1)) which is equal to (n–1)x(n–1).

The elements $\mathbf{K_B}$ have to be so selected that the rank of $(\mathbf{K_B} * \mathbf{K_B^T})$ is (n–1) and that it has mod inverse. Then we have

$$(\mathbf{K_B} * \mathbf{K_B^T})^{-1} = \mathbf{MatModInv}((K_B * K_B^T), p) \quad (17)$$

Post multiplying (16) by $(\mathbf{K_B} * \mathbf{K_B^T})^{-1}$ gives

$$\mathbf{M} = \mathbf{W} * \mathbf{K_B^T} * (\mathbf{K_B} * \mathbf{K_B^T})^{-1} \quad (18)$$

The term $\mathbf{K_B^T} * (\mathbf{K_B} * \mathbf{K_B^T})^{-1}$ is the *right modular pseudo inverse* of $\mathbf{K_B}$ which is designated by $(\mathbf{K_B})^{\dagger}$ which has been specified in (13). From (13) and (18), Equation (12) follows. The size of $(\mathbf{K_B})^{\dagger}$ is nx(n–1).

Example 2. *The message matrix M, at User A, is taken as,*

$$M = \begin{bmatrix} 9 & 10 & 10 \\ 9 & 7 & 6 \\ 10 & 6 & 2 \end{bmatrix}$$

Other matrices and p are same as in Example 1. W *and* $(\mathbf{K_B} * \mathbf{K_B^T})$ *are calculated and found to be,*

$$W{=}M{*}V{*}A = \begin{bmatrix} 8 & 16 & 3 & 9 \\ 10 & 21 & 20 & 3 \\ 5 & 21 & 13 & 0 \end{bmatrix}$$

$$\mathbf{K_B} * \mathbf{K_B^T} = \begin{bmatrix} 20 & 16 & 8 \\ 16 & 0 & 5 \\ 8 & 5 & 6 \end{bmatrix}$$

$(\mathbf{K_B} * \mathbf{K_B^T})^{-1}$ *and* $(\mathbf{K_B})^{\dagger}$ *as given by (13), are found to be,*

$$(\mathbf{K_B} * \mathbf{K_B^T})^{-1} = \begin{bmatrix} 7 & 12 & 19 \\ 12 & 11 & 17 \\ 19 & 17 & 22 \end{bmatrix}$$

$$(\mathbf{K_B})^{\dagger} = \begin{bmatrix} 3 & 0 & 5 \\ 12 & 16 & 16 \\ 5 & 10 & 6 \\ 6 & 0 & 2 \end{bmatrix}$$

All values are calculated with respect to mod p. Then, M *is recovered using (12) as* $\mathbf{M = W * (K_B)^{\dagger}}$.

6 Proxy Re-encryption

Proxy re-encryption [4] is the process of re-encoding a given cipher text so that now, it can be decoded by another receiver other than the original one. The process is so designed that the re-encrypter itself cannot recover the plain text or it can not get hold of the private keys of the concerned parties.

Consider the model shown in Figure 2. Here, **A**, **B** and **C** are the private keys of User A, User B and User C respectively. Matrices **A**, **B**, **G**, **U**, **V**, $\mathbf{K_A}$, $\mathbf{K_B}$, **M** and **W** are same as described in Section 4. **C** is a circulant matrix of size nxn.

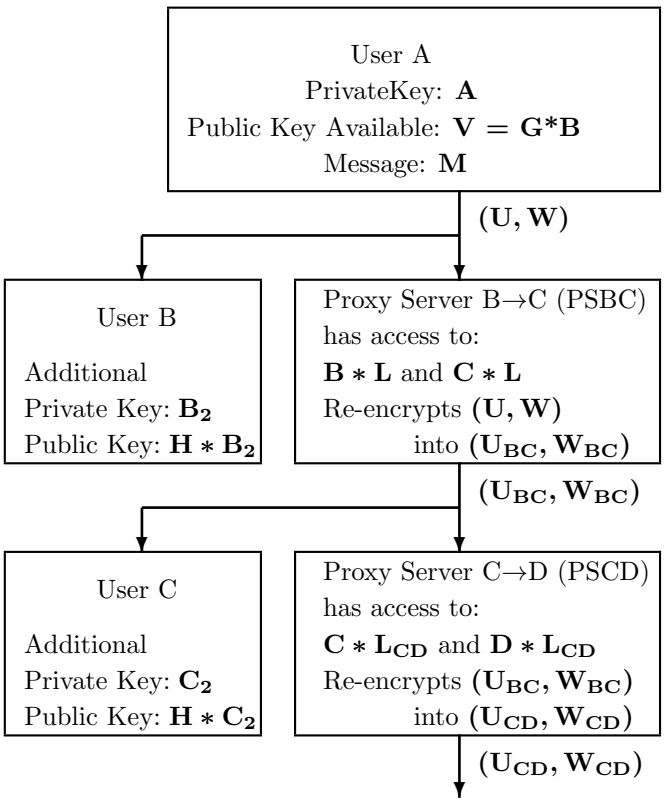


Figure 2: Proxy Re-encryption and decryption

6.1 Common Secret Key between User B and User C.

Here, the DH type common secret key between User B and User C is designed to have the size of nx(n+1) which is bigger compared to that of $\mathbf{K_A}$ or $\mathbf{K_B}$. The reason for using this bigger size is explained later in this section. This bigger common key is derived as follows.

We select one more public generator matrix **H**. The size of **H** is chosen as nx(n+1) and its elements belong to Zp. User B and User C select additional private circulant matrices (keys) $\mathbf{B_2}$ and $\mathbf{C_2}$ of size (n+1)x(n+1). The corresponding public keys are $\mathbf{H * B_2}$ and $\mathbf{H * C_2}$ respectively. By knowing **H** and $\mathbf{H * B_2}$, we cannot determine $\mathbf{B_2}$ because **H** has no left inverse. Similarly, $\mathbf{C_2}$ cannot be determined by knowing **H** and $\mathbf{H{*}C_2}$.

On receiving $\mathbf{H * B_2}$, User C calculates the secret common key between User C and User B as:

$$\mathbf{L_C = (H * B_2) * C_2} \quad (19)$$

Similarly, User B calculates the secret common key between User B and User C as:

$$\mathbf{L_B = (H * C_2) * B_2} \quad (20)$$

Since $\mathbf{B_2}$ and $\mathbf{C_2}$ are multiplicatively commutative $\mathbf{B_2 * C_2 = C_2 * B_2}$ and then, $\mathbf{L_B}$ and $\mathbf{L_C}$ are equal as:

$$\mathbf{L = L_B = L_C = H * C_2 * B_2 = H * B_2 * CB_2} \quad (21)$$

The size of $\mathbf{L}$ is nx(n+1). The matrices $\mathbf{H}$, $\mathbf{B_2}$ and $\mathbf{C_2}$ are so chosen that the rank of $\mathbf{L}$ is n.

6.2 Encryption by User A

Encryption by User A is same as described in Section 4. User A generates the two matrices $\mathbf{U}$ and $\mathbf{W}$ (See (1) and (10)) as repeated here.

$$\begin{aligned} \mathbf{U} &= \mathbf{G} * \mathbf{A} \\ \mathbf{W} &= \mathbf{M} * \mathbf{V} * \mathbf{A} = \mathbf{M} * \mathbf{G} * \mathbf{B} * \mathbf{A} \\ &= \mathbf{M} * \mathbf{G} * \mathbf{A} * \mathbf{B}. \end{aligned}$$

User A sends the encrypted data $\mathbf{(U, W)}$ to User B and to the proxy server B→C. From (4) and (12),

$$\mathbf{M} = \mathbf{W} * (\mathbf{U} * \mathbf{B})^{\dagger} \tag{22}$$

User B can decrypt cipher data $\mathbf{(U, W)}$ as given by Equation (22).

6.3 Re-encryption at Proxy Server B→C

Now the Proxy Server B→C (PSBC) is requested by User A (or by User B) to send the same data $\mathbf{M}$ to User C with proper re-encryption so that User C can decode it correctly. Here, PSBC has to translate the cipher text meant for User B to a new format so that the translated cipher text can be decoded by User C. Basically the PSBC accepts $\mathbf{(U, W)}$ as the input and re-encrypts it to generate ($\mathbf{U_{BC}}$, $\mathbf{W_{BC}}$) which is sent to User C. The re-encrypted term $\mathbf{W_{BC}}$ is so constructed that it can be decoded by User C only. Also, PSBC itself should be incapable of recovering $\mathbf{M}$, $\mathbf{B}$ or $\mathbf{C}$. During initialization, User B sends $\mathbf{(B * L)}$ to PSBC and similarly User C sends $\mathbf{(C * L)}$ to PSBC. The size of $\mathbf{(B * L)}$ as well as $\mathbf{(C * L)}$ is nx(n+1).

6.4 Formulation of $\mathbf{U_{BC}}$ and $\mathbf{W_{BC}}$ at PSBC

For the purpose of re-encryption, PSBC formulates $\mathbf{U_{BC}}$ and $\mathbf{W_{BC}}$ from $\mathbf{U}$ and $\mathbf{W}$ as,

$$\mathbf{U_{BC}} = \mathbf{U} = \mathbf{G} * \mathbf{A} \tag{23}$$

$$\mathbf{W_{BC}} = \mathbf{W} * (\mathbf{C} * \mathbf{L}) * (\mathbf{B} * \mathbf{L})^{\dagger}. \tag{24}$$

Here, $\mathbf{(B * L)}^{\dagger}$ is the right modular inverse of $\mathbf{(B * L)}$. The size of $\mathbf{(B * L)}$ is nx(n+1). By definition,

$$(\mathbf{B} * \mathbf{L})^{\dagger} = (\mathbf{B} * \mathbf{L})^{\mathbf{T}} * ((\mathbf{B} * \mathbf{L}) * (\mathbf{B} * \mathbf{L})^{\mathbf{T}})^{-1} \tag{25}$$

Reducing the parantheses on the RHS of (25), we get,

$$(\mathbf{B} * \mathbf{L})^{\dagger} = \mathbf{L}^{\mathbf{T}} * \mathbf{B}^{\mathbf{T}} * (\mathbf{B} * \mathbf{L} * \mathbf{L}^{\mathbf{T}} * \mathbf{B}^{\mathbf{T}})^{-1} \tag{26}$$

Taking the inverse operator on the RHS of (26) inside the paranthesis we get,

$$(\mathbf{B} * \mathbf{L})^{\dagger} = \mathbf{L}^{\mathbf{T}} * \mathbf{B}^{\mathbf{T}} * (\mathbf{B}^{\mathbf{T}})^{-1} * (\mathbf{L} * \mathbf{L}^{\mathbf{T}})^{-1} * \mathbf{B}^{-1} \tag{27}$$

Now, the RHS of Equation (27) is simplified as,

$$(\mathbf{B} * \mathbf{L})^{\dagger} = \mathbf{L}^{\mathbf{T}} * (\mathbf{L} * \mathbf{L}^{\mathbf{T}})^{-1} * \mathbf{B}^{-1} \tag{28}$$

By definition,

$$\mathbf{L}^{\mathbf{T}} * (\mathbf{L} * \mathbf{L}^{\mathbf{T}})^{-1} = \mathbf{L}^{\dagger} \tag{29}$$

from (28) and (29),

$$(\mathbf{B} * \mathbf{L})^{\dagger} = \mathbf{L}^{\dagger} * \mathbf{B}^{-1} \tag{30}$$

substituting this in (24), we get,

$$\mathbf{W_{BC}} = \mathbf{W} * (\mathbf{C} * \mathbf{L}) * \mathbf{L}^{\dagger} * \mathbf{B}^{-1} \tag{31}$$

On cancelling $\mathbf{L} * \mathbf{L}^{\dagger}$ in (31), we have,

$$\mathbf{W_{BC}} = \mathbf{W} * \mathbf{C} * \mathbf{B}^{-1} \tag{32}$$

Since C and $\mathbf{B}^{-1}$ are circulant matrices, they are multiplicatively commutative. Therefore,

$$\mathbf{C} * \mathbf{B}^{-1} = \mathbf{B}^{-1} * \mathbf{C} \tag{33}$$

From (32) and (33),

$$\mathbf{W_{BC}} = \mathbf{W} * \mathbf{B}^{-1} * \mathbf{C} \tag{34}$$

On substituting for $\mathbf{W}$ from (10) in (34), we get,

$$\mathbf{W_{BC}} = (\mathbf{M} * \mathbf{G} * \mathbf{A} * \mathbf{B}) * \mathbf{B}^{-1} * \mathbf{C} \tag{35}$$

On cancelling $\mathbf{B} * \mathbf{B}^{-1}$ in (35) we have,

$$\mathbf{W_{BC}} = \mathbf{M} * \mathbf{G} * \mathbf{A} * \mathbf{C} \tag{36}$$

In this way the private key $\mathbf{B}$ of User B is eliminated from $\mathbf{W_{BC}}$ and the private key $\mathbf{C}$ is inserted in its place. Now $\mathbf{W_{BC}}$ is ready for decryption by User C. The Proxy Server PSBC sends($\mathbf{U_{BC}}$, $\mathbf{W_{BC}}$) pair to User C.

6.5 Decryption at User C

Once, User C receives $\mathbf{U_{BC}}$ and $\mathbf{W_{BC}}$, $\mathbf{M}$ is recovered from $\mathbf{W_{BC}}$, based on (36). From (36) and (23),

$$\mathbf{W_{BC}} = \mathbf{M} * (\mathbf{U_{BC}}) * \mathbf{C} \tag{37}$$

Therefore $\mathbf{M}$ is recovered by User C as,

$$\mathbf{M} = \mathbf{W_{BC}} * (\mathbf{U_{BC}} * \mathbf{C})^{\dagger} \tag{38}$$

Here, $\mathbf{(U_{BC} * C)}^{\dagger}$ is the right modular inverse of $\mathbf{(U_{BC} * C)}$. It can be seen that (43) is similar to (22). A numerical example of proxy re-encryption is given below.

Example 3. *The values of* ***G, A, B, U, V, M, W*** *and* p *are same as in Example 2. The values of* ***C, H, B2,*** *and* ***C2*** *are taken as,*

$$\boldsymbol{C} = \begin{bmatrix} 9 & 15 & 9 & 6 \\ 6 & 9 & 15 & 9 \\ 9 & 6 & 9 & 15 \\ 15 & 9 & 6 & 9 \end{bmatrix}$$

$$\boldsymbol{H} = \begin{bmatrix} 2 & 9 & 7 & 8 & 18 \\ 15 & 9 & 14 & 11 & 19 \\ 2 & 16 & 5 & 4 & 9 \\ 14 & 7 & 22 & 1 & 15 \end{bmatrix}$$

$$\boldsymbol{B_2} = \begin{bmatrix} 13 & 21 & 3 & 5 & 17 \\ 17 & 13 & 21 & 3 & 5 \\ 5 & 17 & 13 & 21 & 3 \\ 3 & 5 & 17 & 13 & 21 \\ 21 & 3 & 5 & 17 & 13 \end{bmatrix}$$

$$\boldsymbol{C_2} = \begin{bmatrix} 9 & 12 & 8 & 1 & 2 \\ 2 & 9 & 12 & 8 & 1 \\ 1 & 2 & 9 & 12 & 8 \\ 8 & 1 & 2 & 9 & 12 \\ 12 & 8 & 1 & 2 & 9 \end{bmatrix}$$

$\boldsymbol{L}$ *is calculated from (21) as,*

$$\boldsymbol{L} = \begin{bmatrix} 4 & 21 & 4 & 9 & 4 \\ 14 & 20 & 9 & 17 & 7 \\ 1 & 20 & 9 & 15 & 4 \\ 9 & 22 & 1 & 10 & 7 \end{bmatrix}$$

$\mathbf{B} * \mathbf{L}$ *is found to be,*

$$\mathbf{B} * \mathbf{L} = \begin{bmatrix} 2 & 20 & 20 & 11 & 1 \\ 8 & 3 & 19 & 18 & 20 \\ 20 & 1 & 1 & 1 & 1 \\ 13 & 9 & 6 & 13 & 20 \end{bmatrix}$$

$(\mathbf{B} * \mathbf{L})^{\dagger}$ *and* $\mathbf{C} * \mathbf{L}$ *are found to be,*

$$(\mathbf{B} * \mathbf{L})^{\dagger} = \begin{bmatrix} 14 & 11 & 8 & 8 \\ 18 & 19 & 11 & 13 \\ 11 & 16 & 20 & 12 \\ 7 & 20 & 6 & 6 \\ 6 & 1 & 11 & 16 \end{bmatrix}$$

$$\mathbf{C} * \mathbf{L} = \begin{bmatrix} 10 & 19 & 5 & 2 & 12 \\ 16 & 22 & 19 & 16 & 3 \\ 11 & 14 & 2 & 8 & 12 \\ 20 & 8 & 20 & 8 & 3 \end{bmatrix}$$

$\mathbf{W_{BC}}$ *and* $(\mathbf{U_{BC}} * \mathbf{C})$ *are found to be,*

$$\mathbf{W_{BC}} = \begin{bmatrix} 7 & 18 & 9 & 18 \\ 14 & 10 & 3 & 5 \\ 7 & 22 & 13 & 22 \end{bmatrix}$$

$$\mathbf{U_{BC}} * \mathbf{C} = \begin{bmatrix} 21 & 21 & 10 & 20 \\ 17 & 11 & 3 & 5 \\ 20 & 11 & 5 & 11 \end{bmatrix}$$

Matrix $(\mathbf{U_{BC}} * \mathbf{C})^{\dagger}$ *is found to be,*

$$(\mathbf{U_{BC}} * \mathbf{C})^{\dagger} = \begin{bmatrix} 2 & 9 & 18 \\ 8 & 13 & 9 \\ 10 & 9 & 14 \\ 11 & 0 & 21 \end{bmatrix}$$

Finally, $\mathbf{W_{BC}} * (\mathbf{U_{BC}} * \mathbf{C})^{\dagger}$ *is calculated to get* $\boldsymbol{M}$ *as,*

$$\mathbf{W_{BC}} * (\mathbf{U_{BC}} * \mathbf{C})^{\dagger} = \mathbf{M} = \begin{bmatrix} 9 & 10 & 10 \\ 9 & 7 & 6 \\ 10 & 6 & 2 \end{bmatrix}$$

6.6 Re-encryption from User C to User D

The Proxy Server Now the Proxy Server C→D (PSCD) accepts ($\mathbf{U_{BC}}$, $\mathbf{W_{BC}}$) and generates ($\mathbf{U_{CD}}$, $\mathbf{W_{CD}}$) using the similar process as described in Section 6.4. The proxy server should have access to $\mathbf{(C * L_{CD})}$ and $\mathbf{(D * L_{CD})}$ from User C and User D respectively. ($\mathbf{L_{CD}}$ is the common key between User C and User D similar to as given in (21). The decryption at User D would be similar as described in Section 6.5. The above chaining action can be continued further.

7 Discussion

7.1 Matrix Keys Versus Scalar Keys in Cryptography

When scalar keys are used their lengths have to be relatively very large, like 1024 bits, 2048 bits etc.as in RSA. The exponentiation and related arithmetic operations, using these long keys, are more difficult to implement in the processors used in wireless sensor nodes. The processors within the sensor nodes here have limited register sizes and less memory compared to the conventional processors. Therefore, long scalar keys are not convenient for cryptography involving sensor nodes. When integer matrices are used as keys, the length of the individual elements can be kept as low as 8 bits. The large number of elements in a matrix key make hacking very difficult and this very large number will compensate the short length of the key elements. The effective key length of a matrix of size nxn is nxnxm where m is the length of the individual elements of the matrix in bits. The Arithmetic operations on these small sized integers of the matrix keys are easy to implement in the processors of the sensor nodes. Therefore, matrix keys are well suited for the cryptographic operations involving sensor nodes.

7.2 Vulnerabilities of Circulant Matrices in Cryptography

In a circulant matrix, only the first row is chosen independently. The remaining rows are obtained by the circular shift of the preceding rows. Thus a hacker has to break only one row of the circulant key matrix to discover the entire matrix. Thus the effective key length of a circulant matrix used as a key is, nxm where n is number of columns of the matrix and m is the length of the individual elements in bits. Therefore the size of the circulant matrix itself has to be large to provide a large effective

key length. One major advantage is, while storing a circulant matrix, it is enough if we store only the first row. Thus the memory space is saved.

7.3 Comparison with Existing Methods

The closest method to ours is by Keith R Slavin [20], US patent US 7346162 B2. In that work also, the author uses closed Linear Group of matrices **GL**(n, p)'s. But multiplicatively commutative matrices are obtained in a different way other than using the circulant matrices. Our method is quite different from that of [20]. Our scheme is substantially better because, while generating public keys or the shared secret key, we use only one matrix multiplication compared to two as in [20]. The method used in [2], overcomes the deficiency of Cayley-Purser Algorithm [7] that uses even values for n, the size of private keys. But in our case, whether n is even or odd, the security of the cryptosystem will not be compromised. In [21], each user has to store two private keys which are mutually commutative and the number of matrix multiplications is more compared to our method. As far as our knowledge, no earlier work was found on proxy re-encryption using commutative matrices.

8 Conclusions

A new method of Diffie-Hellman type key exchange using circulant matrices as private and public keys is presented. Matrices as keys provide a large number of smaller sized integer elements which are easy to manipulate than a few very large sized integers. Circulant matrices are also used as keys to provide multi-stage proxy re-encryption. Since we use modular multiplication of matrices rather than modular exponentiation, our method is faster and less complex.

References

[1] H. Abdalla, X. Hu, A. Wahaballa, *et al.*, "Integrating the functional encryption and proxy re-cryptography to secure DRM scheme," *International Journal of Network Security*, vol. 19, pp. 27–38, 2017.

[2] G. Ateniese, K. Fu, M. Green and S.Hohenberger, "Improved proxy re-encryption schemes with applications to secure distributed storage," *ACM Transactions on Information and System Security (TISSEC'06)*, vol. 9, no. 1, pp. 1–30, 2006.

[3] M. Blaze, G. Bleumer and Martin Strauss, "Divertible protocols and atomic proxy cryptography," *EUROCRYPT*, vol. 1403, no. 1, pp. 127–144, 1998.

[4] P. J. Davis, *Circulant Matrices (2ed)*, New York: Chelsea Publishing, 2012.

[5] W. Diffie and M. Hellman, "New directions in cryptography," *IEEE Transactions on Information Theory*, vol. 22, no. 6, pp. 644–654, 1976.

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

[7] S. Flannery and D. Flannery, *In code*, New York: Workman publishing, 2001.

[8] K. C. Gupta and I. G. Ray, "On constructions of involutory MDS matrices," in *International Conference on Cryptology in Africa (AFRICACRYPT'13)*, pp. 43–60, June 2013.

[9] L. S. Hill, "Cryptography in an algebraic alphabet," *The American Mathematical Monthly*, vol. 36, no. 6, pp. 306–312, 1929.

[10] S. Inam and R. Ali, "A new ElGamal-like cryptosystem based on matrices over groupring," *Neural Computing and Applications*, pp. 1–5, 2016.

[11] C. H. Ling, S. M. Chen, and M. S. Hwang, "Cryptanalysis of Tseng-Wu group key exchange protocol," *International Journal of Network Security*, vol. 18, no. 3, pp. 590-593, 2016.

[12] J. Nakahara Jr. and E. Abrahao, "A new involutory MDS matrix for the AES," *International Journal of Network Security*, vol. 9, no. 2, pp. 109–116, 2009.

[13] I. Kra and S. R. Simanca, "On circulant matrices, notes," *American Mathematical Society*, vol. 59, no. 3, pp. 368–377, 2012.

[14] C. Lan, H. Li, S. Yin and L. Teng, "A new security cloud storage data encryption scheme based on identity proxy re-encryption," *International Journal of Network Security*, vol. 19, no. 5, pp. 804–810, 2017.

[15] A. Mahalanobis, "The discrete logarithm problem in the group of non-singular circulant matrices," *Groups Complexity Cryptology*, vol. 2, pp. 83–89, 2010.

[16] J. Overbey, W. Traves and J. Wojdylo, "On the keyspace of the hill cipher," *Cryptologia*, vol. 29, pp. 59–72, 2005.

[17] A. V. Ramakrishna and T. V. N. Prasanna, "Symmetric circulant matrices and publickey cryptography," *International Journal of Contemporary Maths and Sciences*, vol. 8, no. 12, pp. 589–593, 2013.

[18] K. A. Reddy, B. Vishnuvardhan, Madhuviswanatham and A. V. N. Krishna, "A modified hill cipher based on circulant matrices," *Procedia Technology*, vol. 4, pp. 114–118, 2002.

[19] M. K. Singh, "Public key cryptography with matrices," *Groups Complexity Cryptology*, vol. 2, pp. 83–89, 2010.

[20] K. R. Slavin, *Public key cryptography using matrices*, US Patent 7346162 B2, 2008.

[21] W. Stallings, *Cryptography and Network Security (4ed)*, India: Pearson Education, 2011.

[22] Y. Wang, D. Yan, F. Li and H. Xiong, "A key-insulated proxy re-encryption scheme for data sharing in a cloud environment," *International Journal of Network Security*, vol. 19, no. 4, pp. 623–630, 2017.

[23] T. Yoshida and M. Shirase, "A digital content sharing model using proxy re-encryption without server access," in *IEEE International Conference*

on Consumer Electronics - Taiwan (ICCE-TW'17), pp. 243–244, 2017.

Biography

Chitra Rajarama received her B.E., in Computer Science and Engineering., and M.Tech., in Computer Science and Engineering, from Visvesvaraya Technological University, Belgaum, Karnataka, India. She is currently pursuing PhD under Visvesvaraya Technological University, Belgaum, Karnataka, India. Her area of interest is in the field of wireless networks. She has guided many undergraduate projects. She has attended many national/international conferences and published several papers in international journals. At present she is Associate Professor in the department of Information Science and Engineering, NIE Institute of Technology, (affiliated to Visvesvaraya Technological University) Mysuru, Karnataka, India.

S. N. Jagadeesha received his B.E.,in Electronics and Communication Engineering, from University B. D. T. College of Engineering., Davangere affiliated to Mysore University, Karnataka, India in 1979, M.E. from Indian Institute of Science (IISC), Bangalore, India specializing in Electrical Communication Engineering., in 1987 and Ph.D. in Electronics and Computer Engineering., from University of Roorkee, Roorkee, India in 1996. He is an IEEE member. His research interest includes Array Signal Processing, Wireless Sensor Networks and Mobile Communications. He has published and presented many papers on Adaptive Array Signal Processing and Direction-of-Arrival estimation. Currently he is professor and head in the department of Computer Science and Engineering, PES Institute of Technology and Management,. (Affiliated to Visvesvaraya Technological University), Shimoga, Karnataka, India.

T. Yerri Swamy received his B.E.,in Electronics and Comm-unication Engineering, from Gulbarga Univerisity,Gulbarga, Karnataka , India in 2000. MTech in Network and Internet Engineering, from Visve-svaraya Technological University, Belgaum, Karnataka. at J. N. N. College of Engineering, Shimoga, Karnataka in 2005. and PhD in the Faculty of Computer and Information Sciences from Visvesvaraya Technological niversity, Belgaum, Karnataka in the year 2013. He is an ISTE member. His research interest includes Antenna Array Signal Processing, Statistical Signal Processing, Detection and Estimation, Cognitive radio comm-unications ,LTE/MIMO. He has published and presented number of papers in national/international conferences and journals. Currently he is Professor and Head, in the department of Computer Science and Engineering, KLE Institute of Technology, (Affiliated to Visvesvaraya Technological University), Hubli, Karnataka, India.

Performance Analysis of RSA and Elliptic Curve Cryptography

Dindayal Mahto and Dilip Kumar Yadav
(Corresponding author: Dindayal Mahto)

Department of Computer Applications, National Institute of Technology Jamshedpur
Jamshedpur-14, Jharkhand, India
(Email: dindayal.mahto@gmail.com)

(Received Mar. 27, 2017; revised and accepted Sept. 21, 2017 & Nov. 5, 2017)

Abstract

This paper presents a performance study and analysis of two popular public-key cryptosystems: RSA with its two variants, and ECC (Elliptic Curve Cryptography). RSA is considered as the first generation public-key cryptography, which is very popular since its inception while ECC is gaining its popularity recently. Besides studying and analyzing the paper also suggests the supremacy among these cryptosystems based on the experimentation. The paper shows the result of the experimentation performed using these cryptosystems with the different modulus/key sizes recommended by the NIST. The modulus/key sizes are used such as 1024/2048/3072-bit for RSA and 160/224/256-bit for ECC. After experimentation and execution of these cryptosystems, the paper concludes that an ECC-based cryptosystem is better than an RSA or its variants-based cryptosystem, and an ECC based cryptosystem best suits for memory-constrained devices, as an ECC-based cryptosystem requires fewer resources than an RSA-based cryptosystem.

Keywords: Decryption; Elliptic Curve Cryptography; Encryption; Public-Key Cryptography; RSA

1 Introduction

Asymmetric key cryptography or public-key cryptography (PKC) uses two keys mainly a private key and a public key; the private key is used for decryption or signature generation while the public key is used for encryption or signature verification. The PKC gains its popularity by developing two pioneering concepts, the firstly, solving key distribution problem of symmetric key cryptography and, then secondly, providing a digital signature scheme [12, 18, 23]. This type of cryptography is mostly used by all leading social and commercial websites for exchanging keys (*i.e.*, small data) in a secure way, and achieving authenticity, integrity, and non-repudiation services. For example, ECDHE_RSA protocols (Elliptic Curve Diffie-Hellman Key Exchange with RSA) are being used by www.amazon.in, and www.linkedin.com, and ECDHE_ECDSA protocols (ECDHE with Elliptic Curve Digital Signature Algorithm) are being used by www.facebook.com, and www.mail.google.com.

RSA [38] is considered as the defacto standard for the public-key cryptography, while ECC [20, 33, 46] is considered as an alternative to RSA. The security of RSA cryptosystem is based on the Integer Factorization Problem (IFP) and the security of ECC is based on the Elliptic Curve Discrete Logarithm Problem (ECDLP). The main attraction of ECC over RSA is that the best-known algorithm for solving the ECDLP takes full exponential time while to solve the IFP of RSA takes sub-exponential time. The fastest algorithm is known as Pollard's rho algorithm for solving the ECDLP takes full exponential time, which has an expected running time $\sqrt{\pi n}/2$. As on 2003, the largest ECDLP instance solved with Pollard's rho algorithm for an elliptic curve over a 109-bit prime field. The best-known generic integer factoring method is Pollard's general number field sieve (NFS). The heuristic expected run-time needed for the NFS to find a factor of the composite number n is L[n] = [1/3, 1.923]. The largest integer factored using the NFS takes sub-exponential time, is the RSA200, a 200-digit number (665-bit) which was factored in May 2005 [16]. This means that, for the same level of security, significantly smaller parameters can be used in ECC than RSA. For example, to achieve 112-bit of security level, an RSA based cryptosystem needs a key of a size of 2048-bit, while an ECC based cryptosystem needs a key of a size of 224-bit [2] as shown in Table 1 and Figure 1. This paper demonstrates the usage of the algorithms of RSA and ECC between two communicating parties (*i.e.*, Alice and Bob).

This paper is organized as follows. In Section 2, the related works and literature reviews are described. In Section 3, RSA and its variants algorithms are described. In Section 4, ECC algorithm is described. In Section 5, different case studies are stated, in Section 6, a performance analysis of RSA and its two variants with ECC

are mentioned and in Section 7, the conclusion is stated.

Table 1: Key size (NIST recommended) [2]

Security Bits level	RSA	ECC
80	1024	160
112	2048	224
128	3072	256
192	7680	384
256	15360	512

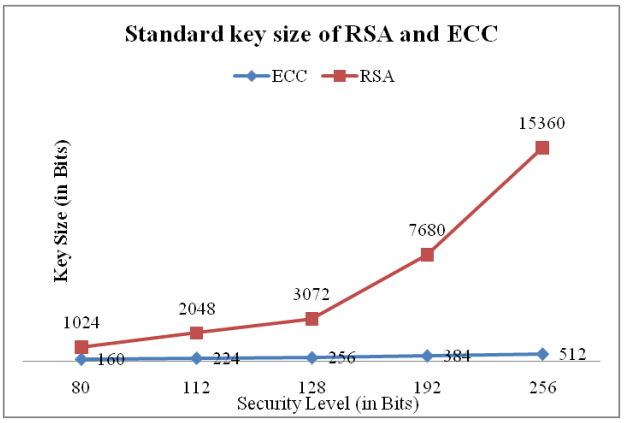


Figure 1: Key size (NIST recommended) [2]

2 Related Works and Literature Reviews

The security/performance analysis of RSA and ECC with different parameters of measurements have been presented by many authors. Gura *et al.* [14] compared point multiplication operation of an elliptic curve over RSA and ECC on two 8-bit processors computer systems and they found on both systems that ECC-160 point multiplication is more efficient than RSA-1024 private-key operation. Bos *et al.* [6] presented an assessment of the risk of the key for RSA and ECC based on key length, and they concluded that till 2014, the use of 1024-bit RSA provides some small risk, while the 160-bit ECC over a prime field may safely be used for a much longer period. Kute *et al.* [21] concluded that RSA is faster but security wise ECC outperforms RSA. Jansma *et al.* [19] compared the usages of digital signatures in RSA and ECC and suggested that RSA may be a good choice for the applications, where verification of a message is required more than a generation of the signature. Alese *et al.* [1] suggested that currently, RSA is stronger than ECC although, however, near future, ECC may outperform RSA. Mahto *et al.* [24–31] demonstrated that ECC outperforms in terms of operational efficiency and security over RSA.

3 RSA and Its Two Popular Variants

Boneh *et al.* [5] presented a survey of four variants of RSA designed to speed up RSA decryption and speed efficiency of these variants using a 1024-bit RSA modulus. They stated that a batch RSA and two multi-factor RSA methods ($n = p^2q$ and n=pqr) are supposed to be fully backward-compatible. They also stated that the rebalanced RSA method provides more speed up with large encryption-exponent 'e'. Chang *et al.* [7] presented a parallel implementation for generating RSA keys using an alternative of the Euclidean Algorithm *i.e.*, Derome's method. The paper claimed the proposed protocol works at low computational cost. Verma *et al.* [47] claimed that modulus and the key generation are achieved using a small order of matrix. In order to generate approximately 840-bit modulus and a private key of RSA, a matrix of four orders is enough. The paper implemented a model in which a small encryption exponent is used to speed up encryption whereas Chinese Remainder Theorem (CRT) is used to speed up decryption time. Ahmad *et al.* [32] proposed a variant of RSA encryption that uses CRT to conceal more than one plaintext in one ciphertext. They proved that their algorithm is safe against several security attacks and proposed some solutions for other security attacks. Santosh *et al.* [41] claimed that they can break the Multi-prime RSA using lattice basis reduction when a user generates 'n' instances with the same modulus. Dong *et al.* [13] proposed to improve threshold secret sharing schemes based RSA with CRT and they claimed that the security channel is not required for their scheme, as each participant chooses his secret shadows by himself as well as the participant can verify the authenticity of secret shadows generated by other participants. Takayasu *et al.* [45] provided enhanced lattice construction for the (δ, β)-SIP and their result shows that if differences of prime factors are small, then the Multi-Prime RSA is vulnerable than the expected.

3.1 RSA (Basic)

RSA (Basic) or RSA [38] is considered as the first real life and practical asymmetric-key cryptosystem. The algorithm (Algorithm 1) for RSA is given below. The security of RSA lies with integer factorization problem.

Here, the key generation is done by each party, once key generation gets over, they can communicate each other securely. In RSA algorithm, for encryption, an exponent e should be chosen such that gcd($\Phi(n)$, e) is equal to 1, and for decryption an exponent, d is generated with the help of finding the inverse of e mod $\Phi(n)$.

In encryption process, the sender has to encrypt the message (*i.e.*, in decimal digit) with the help of the receiver's public key *i.e.*, e and n. In decryption process, the receiver has to decrypt the ciphertext with the help of his own private key *i.e.*, d and n.

Algorithm 1 : RSA (also called RSA (Basic))

RSA algorithm exhibits key generation, encryption, and decryption.

Key Generation

1: Select p, and q; where, p and q both are primes, $p \neq q$.
2: Calculate $n = p \times q$.
3: Calculate $\Phi(n) = (p-1) \times (q-1)$.
4: Select encryption exponent e;
gcd($\Phi(n)$, e) = 1 and (1 <e <$\Phi(n)$).
5: Calculate decryption exponent d;
$d \equiv e^{-1}(mod\ \Phi(n))$.
6: Public key PU = (e, n).
7: Private key PR = (d, n).

Encryption

1: Plaintext: M < n.
2: Ciphertext: $C = M^e$ mod n.

Decryption

1: Ciphertext: C.
2: Plaintext: M=C^d mod n.

3.2 RSA with Chinese Remainder Theorem (CRT)

This method [37] presented a method to break the decryption exponent *i.e.*,d into two parts (d_p, d_q) to decrease the decryption time of RSA, using CRT. Using this technique RSA decryption achieves 4 times faster than RSA (Basic). The algorithm (Algorithm 2) for RSA with CRT is given below.

Algorithm 2 : RSA with CRT

RSA with CRT algorithm exhibits RSA with decryption using CRT.

Key Generation

1: Same as RSA (Basic).

Encryption

1: Same as RSA (Basic).

Decryption

1: Calculate d_p = d mod p-1, and d_q = d mod q-1.
2: Calculate $M_p = C^{d_p}$ mod p, and $M_q = C^{d_q}$ mod q.
3: Calculate M from M_p, and M_q using CRT.

RSA with CRT improves the overall efficiency of RSA.

3.3 Multi-prime RSA

This variant [10] of RSA, further tried to decrease the decryption time with the help of forming modulus 'n' using multiple primes instead of only two primes. It used k primes: p1, p2, . . . , pk. The algorithm (Algorithm 3) for Multi-prime RSA is given below.

Algorithm 3 : Multi-Prime RSA

The Multi-prime RSA algorithm exhibits key generation using multiple primes, encryption, and decryption using CRT.

Key Generation

1: Calculate n = $\prod_{i=1}^{k} p_i$, where, k distinct primes p1, p2, . . . , pk, each one [n/k]-bit in length. For a 1024-bit modulus one can use at most k=3 (*i.e.*, n = pqr).
2: Calculate $\Phi(n) = \prod_{i=1}^{k}(p_i\text{-}1)$.
3: Select e and d as done with RSA (Basic).
4: Calculate $d_i = d\ mod(p_i - 1), where, 1 \leq i \leq k$.
5: Public key PU = (e, n).
6: Private key PR = (d1, d2, . . . , dk).

Encryption

1: Same as RSA (Basic).

Decryption

1: Calculate d_p = d mod p-1, d_q = d mod q-1, and d_r = d mod r-1.
2: Calculate $M_p = C^{d_p}$ mod p, $M_q = C^{d_q}$ mod q, and $M_r = C^{r_q}$ mod r.
3: Calculate M from M_p, M_q, and M_r using CRT.

4 Elliptic Curve Cryptography (ECC)

An ECC over a prime field is defined by following general equation in two variables with coefficients.

$$y^2 = x^3 + ax + b, \tag{1}$$

where, a and b are the coefficient of the elliptic curve, and the discriminant, $\Delta = 4a^3 + 27b^2 \neq 0$. The $\Delta \neq 0$ requires to form a group and hence to implement cryptography using elliptic curve.

An ECC is another promising asymmetric key cryptosystem, independently coined by Miller [33] and Koblitz [20] in the late 1980s. For better and stronger security of data, bigger key sizes require, which means more overhead on the computing systems. Nowadays small devices are playing important role in the digital world, however, these devices have less memory as well as they also require security. In this scenario, RSA becomes second thoughts. An ECC based system is most suitable for memory constraint devices such as Palmtop, Smartphone, Smartcards, etc. For an equivalent level of security, an ECC requires comparatively less or smaller parameters for encryption and decryption than RSA cryptosystem. Bhardwaj *et al.* [4] implemented the algorithms for ECC for point doubling, point addition, scalar multiplication. They also measured the performance of the ElGamal encryption and decryption using Elliptic Curve over a Finite Field. Qian *et al.* [36] did a study of an ECC based Radio Frequency Identification (RFID) security protocol and highlighted some features like, an ECC provides realistic security for communication and tag memory data access, it also reduces the key storage requirement and the back-

end system by storing private key only, the protocol uses XOR, bitwise AND, so forth which further reduces the tag computation, and at the end, the BAN-logic is used to discuss computational performance, security features, and formal proof of the protocol. Basu [3] presented a transformation algorithm that reduces the number of elementary operations, whereas a parallel computation and the concatenation stages reduce the computational cost using elegant parallel implementation. This simulation shows that the speed attains value nearly equal to the order of N, where N is the number of processors. Srinath *et al.* [44] proposed an Undeniable Blind Signature true Scheme (UBSS) based on the features of isogenies between super-singular elliptic curves and proved that their scheme is safe in the presence of a quantum adversary under certain assumptions. Hou *et al.* [17] proposed a robust and efficient remote authentication scheme with the help of an ECC using CAPTCHA technique and provided a formal proof of the scheme using the BAN-logic. Han *et al.* [15] proposed a new authentication scheme to protect user anonymity and insecure against impersonation attack. They compared their scheme with recent schemes and claimed that their scheme can provide stronger security and more efficiency. Naresh *et al.* [34] proposed an ECDLP based dynamic contributory group key agreement protocol for secure group communication over adhoc networks. Liu *et al.* [22] presented that the algebraic structure of bilinear groups loses the advantages of ECC which gains mainly from smaller parameter size and hence they claimed that this structure is not fit for to cryptographic schemes. The algorithm (Algorithm 4) for ECC is given below.

Here, a P_m is an x, y point encoded with the help of a plaintext message, 'm'. This type of different points is used for encryption and decryption in ECC.

This illustration (Algorithm 5) exhibits a data communication security model for an (OTP) One-Time Password (*i.e.*, "32145688") message using an ECC based cryptosystem.

Algorithm 4 : ECC

ECC algorithm exhibits key generation, encryption, and decryption.

Global public elements

1: Chooses an elliptic curve $E_q(a, b)$ with parameters a, b, and q, where q is a prime and > 3, or an integer of the form 2^m.
2: Selects G(x, y) - a global point on elliptic curve whose order is large value n.

Alice key generation

1: Selects a private key, V_A; where, $V_A < n$.
2: Calculates the public key, $P_A(x, y)$;
$P_A(x, y) = V_A \times G(x, y)$.

Bob key generation

1: Selects a private key, V_B; where, $V_B < n$.
2: Calculates the public key, $P_B(x, y)$;
$P_B(x, y) = V_B \times G(x, y)$.

Secret key calculation by Alice

1: $S_K(x, y) = V_A \times P_B(x, y)$.

Secret key calculation by Bob

1: $S_K(x, y) = V_B \times P_A(x, y)$.

Encryption by Alice using public key of Bob

1: Alice chooses message $P_m(x, y)$ and a random positive integer 'k' and $1 < k < q$.
2: Ciphertext, $C_m((x, y), (x, y))$;
$= ((k \times G(x, y)), (P_m(x, y) + k \times P_B(x, y)))$.

Decryption by Bob using his own private key

1: Ciphertext, $C_m((x, y), (x, y))$.
2: Plaintext, $P_m(x, y)$;
$= (P_m(x, y) + k \times P_B(x, y))$ - $(k \times V_B \times G(x, y))$
$= P_m(x, y)$.
Here, first coordinate of C_m gets multiplied with the private key of the Bob *i.e.*, V_B, which in turns becomes similar to Bob's public key. Finally, due to subtraction of resultant coordinate with the second coordinate of the ciphertext C_m, all get canceled and only $P_m(x, y)$ gets left.

This illustration (Algorithm 5) exhibits a data communication security model for an (OTP) One-Time Password (*i.e.*, "32145688") message using an ECC based cryptosystem.

5 Different Case Studies of Implementation of RSA or/and ECC in Software Security, Hardware Security, Wireless LAN Security

5.1 Implementing Software Security

Public-key cryptography provides two important services of information security. They are as follows:

- Secrecy of information: It is provided using encryption and decryption algorithms.
- Authentication of information: It is provided by implementing a digital signature algorithm.

Algorithm 5 : ECC (An illustration of ECC)

The key generation, encryption, and decryption of ECC use a 160-bit modulus and key size.

Global public parameters

1: Consider a prime number q = 526140590075080893144921154061106107001433154304 73, a = 0, b = 2, G(x) = 1, and G(y) = 25169214788133730807822856623907162551920125398 23.
Based on global public parameters, the elliptic curve equation becomes:

$$\begin{aligned} & y^2 \bmod 526140590075080893144921154061106107001433154304 73 \\ = \; & (x^3 + 2) \bmod 526140590075080893144921154061106107001433154304 73. \end{aligned} \tag{2}$$

Alice Key Generation

1: Selects a random private key, V_A; The value of V_A is 123456789.
2: Calculates the public key, $P_A(x, y)$;
$P_A(x, y) = V_A \times G(x, y)$= 123456789 * (1, 25169214788133730807822856623907162551920125398 23) = (41598408633041765117904814957892579387787288121723, 30331027768661217591203589322093387887193929936089).

Bob Key Generation

1: Selects a random private key, V_B; The value of V_B is 987654321.
2: Calculates the public key, $P_B(x, y)$;
$P_B(x, y) = V_B \times G(x, y)$= 987654321 * (1, 25169214788133730807822856623907162551920125398 23) = (16805040787928634191862900600614936768762639911 52, 111209 26205562069510265635750083533321030274383215).

Secret key calculation by Alice

1: $S_K(x, y) = V_A \times P_B(x, y)$= 123456789 * (16805040787928634191862900600614936768762639911 52, 111209 26205562069510265635750083533321030274383215) = (38027124171320004658630075130620602090153656563382, 93589686927972666553334105380507073201512099436 80).

Secret key calculation by Bob

1: $S_K(x, y) = V_B \times P_A(x, y)$= 987654321 * (41598408633041765117904814957892579387787288121723, 30331027768661217591203589322093387887193929936089) = (38027124171320004658630075130620602090153656563382, 93589686927972666553334105380507073201512099436 80).

In this way, both parties get same secret key *i.e.*, $S_K(x, y)$. In this illustration, 1% of the abscissa (*i.e.*, x coordinate) of S_K(x, y) is used in encoding and decoding of points in elliptic curve.

Encryption of plain OTP by Alice using public key of Bob

1: Considers a plain OTP message as 32145688.
2: Encodes the plain message into encoded message points in the elliptic curve using Koblitz algorithm as shown in Table 2 and in Figure 2.
3: Encrypts the encoded message points into cipher message points as shown in Table 3 and in Figure 3, and sends the cipher message points to Bob.

Decryption by Bob using his own private key

1: Decrypts cipher message points into encoded message points as shown as in Table 2 and in Figure 2.
2: Decodes the encoded points into a plain message.
3: Gets a plain message as 32145688.

Table 2: Encoded message points in the elliptic curve

SN	Pmsg(X)	Pmsg(Y)
1	1022	397240633960111792419914815431672379357145101291 10
2	1001	35793323117293218357572660594255084988589888412 11
3	981	45449421598805694936939313852002285339988567792348
4	1042	41905338400807894834998928213746564044963682008534
5	1064	30965928523274769546027471014828764333153039292943
6	1087	20058071397683232973900828570537209896964825497709
7	1122	48672724547686170285991948417220911731463609191 09
8	1122	48672724547686170285991948417220911731463609191 09

Table 3: Cipher points in the elliptic curve

SN	Cmsg(X)	Cmsg(Y)
1	3031810543774541201270781189866076098338401111071 5	1156970749190670611742309402469442074729998522372 0
2	5220565192955449651963933922116155050413490429786 8	4071180899410416290074400136769496460076809974938 7
3	4063738526495059017862661246185151900917126904100 8	2126988817572347371992175126181578308018171865621 4
4	4801967098760137765050857433861155600749370514324 8	4636953647781384403536987964554977842709593733263 5
5	3248537397588909901435710360360990140998438888930 9	5031581969545167535118204283802643604979516656307 2
6	3555561870533179327695326780282198676588333029932 7	3656933889631812613860067603913087384355220557261 9
7	1457304168690753913199434478985994693173555234169 1	4868401237653324383034229163974356725854951182154 7
8	1457304168690753913199434478985994693173555234169 1	4868401237653324383034229163974356725854951182154 7

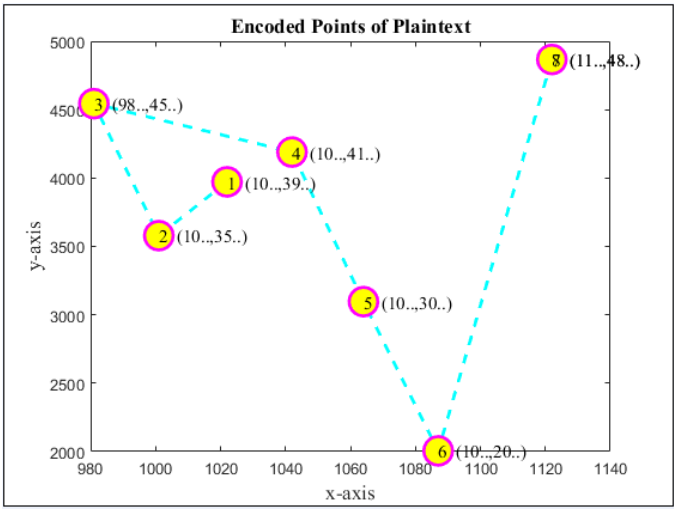


Figure 2: Encoded plain OTP before encryption

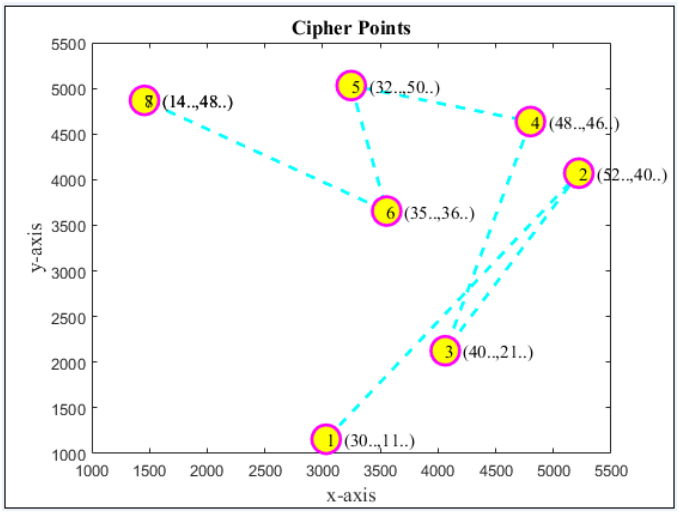


Figure 3: Cipher OTP points

5.2 Secrecy of Information

Case Study 1: Multi-authority Electronic Voting Scheme Based on Elliptic Curves by Porkodi *et al.* [35]. This paper proposed a security model for e-voting system, which works better with same parameters as used in DSA for building secured e-voting system. The paper also proposed that ECC needs considerably smaller parameters and provides the equivalent level of security as other asymmetric algorithms RSA and DSA which need much larger keys.

Case Study 2: Comparative Analysis of Public-Key Encryption Schemes by Alese *et al.* [1]. This research work focused on the comparative analysis of RSA encryption algorithm, ElGamal Elliptic Curve encryption algorithm, and Menezes-Vanstone elliptic curve encryption algorithm. These elliptic curve encryption schemes analog of ElGamal encryption scheme were implemented in Java, using the classes from the FlexiProvider library for RSA and ECC. Performance evaluation of the three algorithms based on the time lapse for their key generation, encryption, and decryption algorithms, and encrypted data size was carried out and compared. Their result confirmed that elliptic curve-based implementations are more superior to RSA-base implementations on all comparative parameters.

After comparing RSA and ECC ciphers, it has been proved that ECC involves much fewer overheads than RSA. ECC has many advantages due to its ability to provide the same level of security as RSA yet using shorter keys. However, its disadvantage which may even hide its attractiveness is its lack of maturity, as mathematicians, they believed that enough research has not yet been done in ECDLP.

Case Study 3: Nonce based ECC for Text and Image applications by Vigila *et al.* [48]. This paper implemented model based on ECC for text and image applications security. The paper suggested that ECC facilitates such as higher strength per bit leading to faster computation reduced power consumption and fewer storage requirements compared to RSA.

5.3 Authentication of Information

Case Study 1: Performance Comparison of Elliptic Curve and RSA Digital Signatures by Nicholas Jansma *et al.* [19]. This paper compared the performance characteristics of two public key cryptosystems (RSA and ECC) used in digital signatures to determine the applicability of each in modern technological devices and protocols that use such signatures.

Their findings suggest that for RSA key of size 1024-bit and greater, RSA key generation is significantly slower than ECC key generation. RSA is comparable to ECC for digital signature creation in terms of time and is faster than ECC for digital signature verification. Thus, for applications requiring message verification more often than the signature generation, RSA may be the better choice.

Case Study 2: A Secure and Efficient Remote User Authentication Scheme for Multi-server Environments Using ECC by Zhang, Junsong, *et al.* [49]. The paper presented that the requirements of operations are lesser in ECC-based than other related asymmetric-key schemes. That means that ECC requires less computational cost than other related public-key cryptosystems. The demonstration of the paper exhibits that proposed scheme can solve various types of security problems and is better suitable for memory-constrained devices.

Case Study 3: Chang *et al.* [8] proposed a strong RSA-based certificate-less signature scheme and claimed that their scheme is capable of resisting more intense malicious behavior.

Case Study 4: Sharma *et al.* [42] proposed an RSA-based efficient certificate-less signature scheme and proved that their scheme is safe under some well-studied assumptions. They also claimed that their scheme is suitable for WSN based on their implementation results on WSN.

Case Study 5: Deng *et al.* [11] proposed an identity-based proxy ring signature (IBPS) scheme using RSA without pairings, and used the random oracle model to prove the security of their scheme. They claimed that their scheme is more efficient than similar ones developed based on bilinear pairings.

Case Study 6: Singh *et al.* [43] experimentally evaluated the performance of digital signature signing and verification processes using RSA (Basic) and ECC. They claimed that RSA signature signing is slower than verification whereas ECC signature signing is generally faster than verification. This paper suggested that to use of ECC in place of RSA.

5.4 Implementing Hardware Security

Case Study 1: ECCs by Robshaw *et al.* [39]. In their paper, they provided a high-level comparison of RSA public-key cryptosystem and proposals for public-key cryptography based on elliptic curves.

There are however many issues to consider when making the choice between applications based on an elliptic curve cryptosystem and one based on RSA. In the paper, they have presented some of the issues (security, performance, standards and interoperability) that are perhaps most pertinent when making such a choice. The comparisons in this paper are made, however, under the premise that an elliptic curve cryptosystem over $GF(2^{160})$ offers the same security as 1024-bit RSA.

Case Study 2: Comparing ECC and RSA on 8-Bit CPUs by Gura *et al.* [14]. They proposed a new algorithm to reduce the number of memory accesses. Implementation and analysis led to three observations:

1) Public-key cryptography is viable on small devices without hardware acceleration. On an Atmel ATmega128 at 8 MHz, they measured 0.81s for 160-bit ECC point multiplication and 0.43s for a RSA-1024 operation with exponent $e = 2^{16} + 1$.
2) The relative performance advantage of ECC point multiplication over RSA modular exponentiation increases with the decrease in processor word size and the increase in key size.
3) Elliptic curves over fields using pseudo-Mersenne primes as standardized by NIST and SECG allow for high- performance implementations and show no performance disadvantage over optimal extension fields or prime fields selected specifically for a particular processor architecture.

They compared elliptic curve point multiplication over three SECG/NIST curves secp160r1, secp192r1, and secp224r1 with RSA-1024 and RSA-2048 on two 8-bit processor architectures. On both platforms, ECC-160 point multiplication outperforms RSA-1024 private-key operation by an order of magnitude and is a factor of 2 of RSA-1024 public-key operation. They presented a novel multiplication algorithm that significantly reduces the number of memory accesses. This algorithm led to a 25% performance increase for ECC point multiplication on the Atmel AVR platform. Their measurements and analysis led to fundamental observations: The relative performance of ECC over RSA increases as the word size of the processor decrease. This stems from the fact that the complexity of addition, subtraction and optimized reduction based on sparse pseudo-Mersenne primes grows linearly with the decrease of the word size

whereas Montgomery reduction grows quadratically. As a result, ECC point multiplication on small devices becomes comparable in performance to RSA public-key operations and they expect it to be higher for large key sizes.

Case Study 3: Chatterjee *et al.* [9] focused on implementing an efficient architecture for scalar multiplication on binary Edwards curve in an analytical way and based on analytical and experimental results they claimed that their model helped in developing an architecture with improved efficiency in comparison to other similar models.

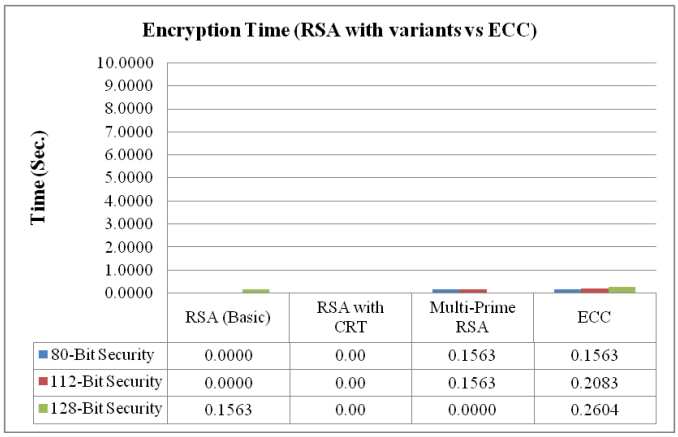
5.5 Wireless LAN Security

Case Study 1: Comparative Performance Analysis of Public-Key Cryptographic Operations in the WTLS Handshake Protocol by Rodriguez-Henriquez *et al.* [40]. They proposed a model for the protocol analysis considering the processing time of the cryptographic operations performed by the Client and the Server during the Negotiation protocol.

In their paper, an efficient realization of the WTLS (Wireless Transport Layer Security) handshake protocol was implemented on a realistic wireless scenario composed of a typical mobile device wirelessly connected to a workstation server. The data gathered in their experiments show that ECC consistently outperforms the traditional option represented by RSA in all the scenarios tested. Additionally, their analytical model predictions show a reasonable agreement with the obtained real data.

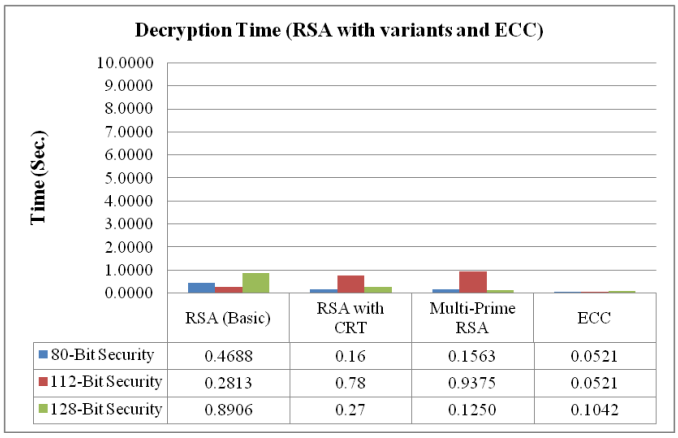
6 Performance Analysis of RSA and Its Two Variants with ECC

A performance analysis, based on encryption, decryption, and total time of RSA (Basic) with its two variants and ECC is mentioned here. The first variant of RSA is RSA with CRT and the second variant of RSA is the Multi-Prime RSA. For measuring time efficiency of these algorithms, the modulus used in experimentation are of 1024/2048/3072-bit for RSA and 160/224/256-bit for ECC, with two sample OTP message data of 27-bit (*i.e.*, "32145688") and 270-bit (*i.e.*, "OTP to transfer money to beneficiary A/C is "34741608". Do not share it with anyone"). Programs for these algorithms written and executed in C with GMP library, on Intel Pentium laptop with a dual-core processor (1.60 GHz, 533 MHz, 1 MB L2 cache), 2GB DDR2 RAM, under Ms-Windows platform. The performance analysis of RSA with its variants over ECC is shown in figures (Figure 4, 5, 6, 7, 8, 9. Upon experimentation, it is found that RSA (Basic) takes more time than its own two variations and ECC. ECC is better in terms of operational efficiency than RSA and its both variants as shown in Figure 6, and Figure 9.



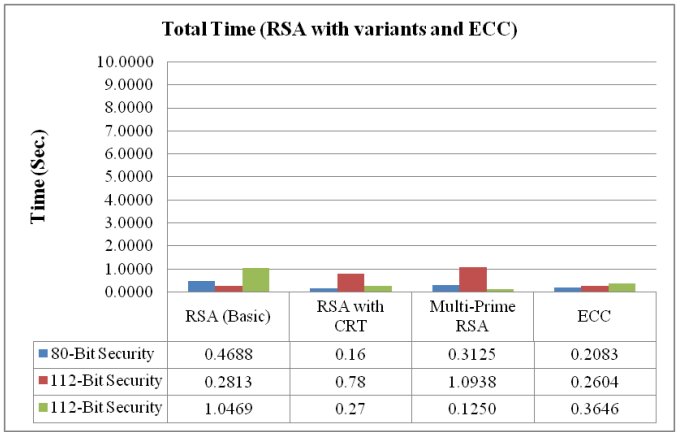
	RSA (Basic)	RSA with CRT	Multi-Prime RSA	ECC
80-Bit Security	0.0000	0.00	0.1563	0.1563
112-Bit Security	0.0000	0.00	0.1563	0.2083
128-Bit Security	0.1563	0.00	0.0000	0.2604

Figure 4: Encryption time (in seconds) of 27-bit data



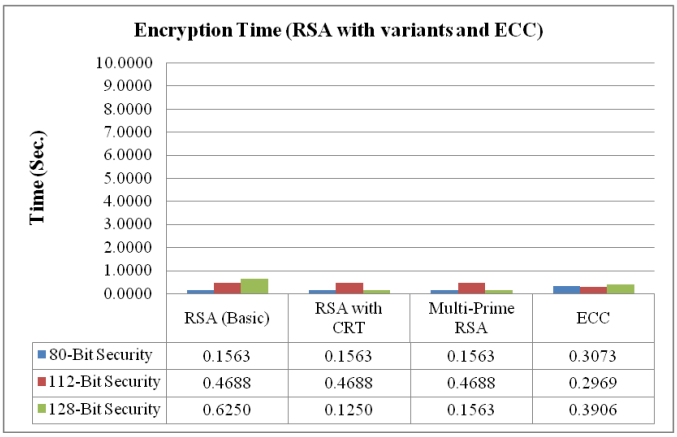
	RSA (Basic)	RSA with CRT	Multi-Prime RSA	ECC
80-Bit Security	0.4688	0.16	0.1563	0.0521
112-Bit Security	0.2813	0.78	0.9375	0.0521
128-Bit Security	0.8906	0.27	0.1250	0.1042

Figure 5: Decryption time (in seconds) of 27-bit data



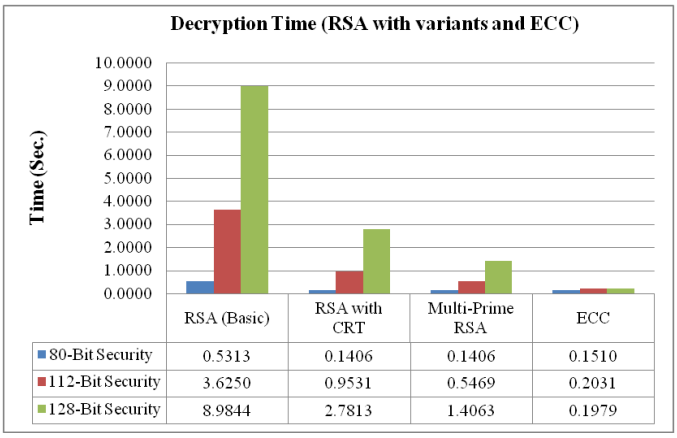
	RSA (Basic)	RSA with CRT	Multi-Prime RSA	ECC
80-Bit Security	0.4688	0.16	0.3125	0.2083
112-Bit Security	0.2813	0.78	1.0938	0.2604
112-Bit Security	1.0469	0.27	0.1250	0.3646

Figure 6: Total (Enc. and Dec.) time (in seconds) of 27-bit data



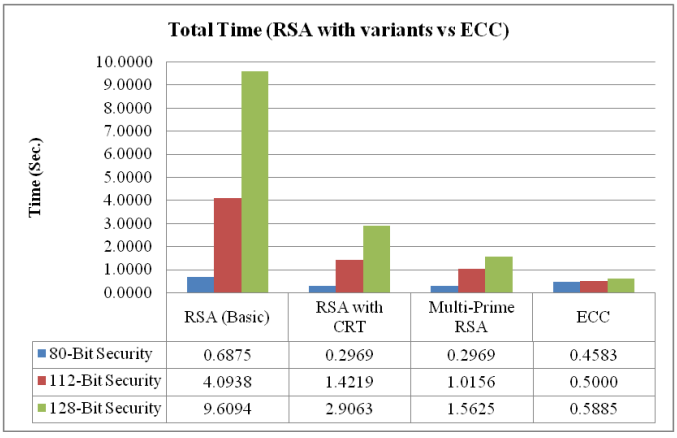
	RSA (Basic)	RSA with CRT	Multi-Prime RSA	ECC
80-Bit Security	0.1563	0.1563	0.1563	0.3073
112-Bit Security	0.4688	0.4688	0.4688	0.2969
128-Bit Security	0.6250	0.1250	0.1563	0.3906

Figure 7: Encryption time (in seconds) of 270-bit data



	RSA (Basic)	RSA with CRT	Multi-Prime RSA	ECC
80-Bit Security	0.5313	0.1406	0.1406	0.1510
112-Bit Security	3.6250	0.9531	0.5469	0.2031
128-Bit Security	8.9844	2.7813	1.4063	0.1979

Figure 8: Decryption time (in seconds) of 270-bit data



	RSA (Basic)	RSA with CRT	Multi-Prime RSA	ECC
80-Bit Security	0.6875	0.2969	0.2969	0.4583
112-Bit Security	4.0938	1.4219	1.0156	0.5000
128-Bit Security	9.6094	2.9063	1.5625	0.5885

Figure 9: Total (Enc. and Dec.) time (in seconds) of 270-bit data

7 Conclusion

Security of data communication is very important while data are being transmitted from one user to another user or system. Cryptography is one of the techniques to provide data communication security. This paper presented a performance study and an analysis of RSA (Basic) with its two popular variants and ECC. The experimental results for encryption, decryption and total time are taken by RSA with its variants and ECC are shown. It is concluded that ECC outperforms RSA and all the mentioned variants of RSA in terms of operational efficiency and security with lesser parameters. ECC with the Affine coordinate system is implemented here, the future research may implement the ECC with other than the Affine coordinate system to improve more efficiency of the ECC.

Acknowledgments

We would like to thank our colleagues, Head of Department of Computer Applications, Dean (R & C) and Director of our Institute for supporting directly or indirectly to this research work.

References

[1] B. K. Alese, E. D. Philemon and S. O. Falaki, "Comparative analysis of public-key encryption schemes," *International Journal of Engineering and Technology*, vol. 2, no. 9, pp. 1552–1568, 2012.

[2] E. Barker, W. Barker, W. Burr, W. Polk and M. Smid, "Recommendation for key management part 1: General (revision 3)," *NIST Special Publication*, vol. 800, no. 57, pp. 1–147, 2012.

[3] S. Basu, "A new parallel window-based implementation of the elliptic curve point multiplication in multi-core architectures," *Group*, vol. 14, pp. 101-108, 2012.

[4] K. Bhardwaj and S. Chaudhary, "Implementation of elliptic curve cryptography in c," *International Journal on Emerging Technologies*, vol. 3, no. 2, pp. 38–51, 2012.

[5] D. Boneh and H. Shacham, "Fast variants of RSA," *CryptoBytes*, vol. 5, no. 1, pp. 1–9, 2002.

[6] J. Bos, M. Kaihara, T. Kleinjung, A. K. Lenstra and P. L. Montgomery, *On the Security of 1024-bit RSA and 160-bit Elliptic Curve Cryptography*, Technical Report, 2009.

[7] C. C. Chang and M. S. Hwang, "Parallel computation of the generating keys for RSA cryptosystems," *Electronics Letters*, vol. 32, no. 15, pp. 1365–1366, 1996.

[8] C. C. Chang, C. Y. Sun and S. C. Chang, "A strong RSA-based and certificateless-based signature scheme," *International Journal Network Security*, vol. 18, no. 2, pp. 201–208, 2016.

[9] A. Chatterjee and I. Sengupta, "Performance modelling and acceleration of binary edwards curve processor on fpgas," *International Journal of Electronics and Information Engineering*, vol. 2, no. 2, pp. 80–93, 2015.

[10] T. Collins, D. Hopkins, S. Langford and M. Sabin, *Public Key Cryptographic Apparatus and Method*, Dec. 8, 1998. US Patent 5,848,159.

[11] L. Deng, H. Huang and Y. Qu, "Identity based proxy signature from RSA without pairings," *International Journal Network Security*, vol. 19, no. 2, pp. 229–235, 2017.

[12] W. Diffie and M. Hellman, "New directions in cryptography," *IEEE Transactions on Information Theory*, vol. 22, no. 6, pp. 644–654, 1976.

[13] X. D. Dong, "A multi-secret sharing scheme based on the crt and RSA," *International Journal of Electronics and Information Engineering*, vol. 2, no. 1, pp. 47–51, 2015.

[14] N. Gura, A. Patel, A. Wander, H. Eberle and S. C. Shantz, "Comparing elliptic curve cryptography and RSA on 8-bit cpus," in *CHES*, vol. 4, pp. 119–132, 2004.

[15] L. Han, Q. Xie and W. Liu, "An improved biometric based authentication scheme with user anonymity using elliptic curve cryptosystem," *International Journal Network Security*, vol. 19, no. 3, pp. 469–478, 2017.

[16] D. Hankerson, A. J. Menezes and S. Vanstone, *Guide to Elliptic Curve Cryptography*, Springer Science & Business Media, 2006.

[17] G. Hou and Z. Wang, "A robust and efficient remote authentication scheme from elliptic curve cryptosystem," *International Journal Network Security*, vol. 19, no. 6, pp. 904–911, 2017.

[18] M. S. Hwang, K. F. Hwang, I. C. Lin, "Cryptanalysis of the batch verifying multiple RSA digital signatures", *Informatica*, vol. 11, no. 1, pp. 15-19, Jan. 2000.

[19] N. Jansma and B. Arrendondo, "Performance comparison of elliptic curve and RSA digital signatures," *nicj. net/files*, 2004.

[20] N. Koblitz, "Elliptic curve cryptosystems," *Mathematics of computation*, vol. 48, no. 177, pp. 203–209, 1987.

[21] V. B. Kute, P. R. Paradhi and G. R. Bamnote, "A software comparison of RSA and ECC," *International Journal Computer Science Applications*, vol. 2, no. 1, pp. 43–59, 2009.

[22] L. Liu, Z. Cao, W. Kong and J. Wang, "On bilinear groups of a large composite order," *International Journal of Electronics and Information Engineering*, vol. 7, no. 1, pp. 1–9, 2017.

[23] E. J. L. Lu, M. S. Hwang, and C. J. Huang, "A new proxy signature scheme with revocation", *Applied Mathematics and Computation*, vol. 161, no. 3, PP. 799-806, Feb. 2005.

[24] D. Mahto, D. A. Khan and D. K. Yadav, "Security analysis of elliptic curve cryptography and RSA," in *Proceedings of the World Congress on Engineering*, vol. 1, 2016.

[25] D. Mahto and D. K. Yadav, "Network security using ecc with biometric," in *International Conference on Heterogeneous Networking for Quality, Reliability, Security and Robustness*, pp. 842–853, 2013.

[26] D. Mahto and D. K. Yadav, "Enhancing security of one-time password using elliptic curve cryptography with biometrics for e-commerce applications," in *Third International Conference on Computer, Communication, Control and Information Technology (C3IT'15)*, pp. 1–6, 2015.

[27] D. Mahto and D. K. Yadav, "Enhancing security of one-time password using elliptic curve cryptography with finger-print biometric," in *2nd International Conference on Computing for Sustainable Global Development (INDIACom'15)*, pp. 1737–1742, 2015.

[28] D. Mahto and D. K. Yadav, "Security improvement of one-time password using crypto-biometric model," in *Proceedings of 3rd International Conference on Advanced Computing, Networking and Informatics*, pp. 347–353, 2016.

[29] D. Mahto and D. K. Yadav, "One-time password communication security improvement using elliptic curve cryptography with iris biometric," *International Journal of Applied Engineering Research*, vol. 12, no. 18, pp. 7105–7114, 2017.

[30] D. Mahto and D. K. Yadav, "Rsa and ECC: A comparative analysis," *International Journal of Applied Engineering Research*, vol. 12, no. 19, pp. 9053–9061, 2017.

[31] D. Mahto and D. K. Yadav, "Secure online medical consultations using elliptic curve cryptography with iris biometric," *International Journal of Control Theory and Applications*, vol. 10, no. 13, pp. 169–179, 2017.

[32] A. Mansour, A. Davis, M. Wagner, R. Bassous, H. Fu and Y. Zhu, "Multi-asymmetric cryptographic RSA scheme," in *Proceedings of the 12th Annual Conference on Cyber and Information Security Research*, p. 9, 2017.

[33] V. S. Miller, "Use of elliptic curves in cryptography," in *Conference on the Theory and Application of Cryptographic Techniques*, pp. 417–426, 1985.

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

[35] C. Porkodi, R. Arumuganathan and K. Vidya, "Multi-authority electronic voting scheme based on elliptic curves.," *International Journal Network Security*, vol. 12, no. 2, pp. 84–91, 2011.

[36] Q. Qian, Y-L Jia and R. Zhang, "A lightweight rfid security protocol based on elliptic curve cryptography," *International Journal Network Security*, vol. 18, no. 2, pp. 354–361, 2016.

[37] J. J. Quisquater and C. Couvreur, "Fast decipherment algorithm for RSA public-key cryptosystem," *Electronics letters*, vol. 18, no. 21, pp. 905–907, 1982.
[38] R. L. Rivest, A. Shamir and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," *Communications of the ACM*, vol. 21, no. 2, pp. 120–126, 1978.
[39] M. J. B. Robshaw and Y. L. Yin, "Elliptic curve cryptosystems," *An RSA Laboratories Technical Note*, vol. 1, p. 997, 1997.
[40] F. Rodrguez-Henrquez, C. E. Lpez-Peza, M. A. Len-Chvez and P. Puebla, "Comparative performance analysis of public-key cryptographic operations in the wtls handshake protocol," in *Proceedings of the 1st International Conference on Electrical and Electronics Engineering*, pp. 24–27, 2004.
[41] K. R. Santosh, C. Narasimham and P. Shetty, "Cryptanalysis of multi-prime RSA with two decryption exponents," *International Journal of Electronics and Information Engineering*, vol. 4, no. 1, pp. 40–44, 2016.
[42] G. Sharma, S. Bala and A. K. Verma, "An improved RSA-based certificateless signature scheme for wireless sensor networks," *International Journal Network Security*, vol. 18, no. 1, pp. 82–89, 2016.
[43] S. R. Singh, A. K. Khan and S. R. Singh, "Performance evaluation of RSA and elliptic curve cryptography," in *2nd International Conference on Contemporary Computing and Informatics (IC3I'16)*, pp. 302–306, 2016.
[44] M. S. Srinath and V. Chandrasekaran, "Isogeny-based quantum-resistant undeniable blind signature scheme," *International Journal of Network Security*, vol. 20, no. 1, pp. 8–17, 2018.
[45] A. Takayasu and N. Kunihiro, "General bounds for small inverse problems and its applications to multi-prime RSA," *IEICE TRANSACTIONS on Fundamentals of Electronics, Communications and Computer Sciences*, vol. 100, no. 1, pp. 50–61, 2017.
[46] S. F. Tzeng, M. S. Hwang, "Digital signature with message recovery and its variants based on elliptic curve discrete logarithm problem", *Computer Standards & Interfaces*, vol. 26, no. 2, pp. 61–71, Mar. 2004.
[47] P. Verma, D. Mahto, S. K. Jha and D. K. Yadav, "Efficient RSA cryptosystem with key generation using matrix," *International Journal of Control Theory and Applications*, vol. 10, no. 13, pp. 221–228, 2017.
[48] S. M. C. Vigila and K. Muneeswaran, "Nonce based elliptic curve cryptosystem for text and image applications.," *International Journal Network Security*, vol. 14, no. 4, pp. 236–242, 2012.
[49] J. Zhang, J. Ma, X. Li and W. Wang, "A secure and efficient remote user authentication scheme for multi-server environments using ECC," *TIIS*, vol. 8, no. 8, pp. 2930–2947, 2014.

Biography

Dindayal Mahto is a Faculty Member cum Ph. D. Research Scholar in the Department of Computer Applications at National Institute of Technology Jamshedpur, India. He received his M. Tech. degree in Information Security from Birla Institute of Technology, Mesra, Ranchi, Jharkhand, India in 2012. His research interests include cryptography, information/biometric security.

Dilip Kumar Yadav is an Associate Professor in the Department of Computer Applications at National Institute of Technology, Jamshedpur, India. He received his Ph. D. degree in software reliability engineering from Indian Institute of Technology, Kharagpur (India) in 2012. He received the B. Tech. (ME) and M. Tech. (CIDM) degrees from National Institute of Technology, Jamshedpur, India, in 1991 and 1994 respectively. His research interests include software reliability and quality modeling, software security, soft computing and system optimization.

Security Improvements of EPS-AKA Protocol

Mourad Abdeljebbar and Rachid El Kouch
(Corresponding author: Mourad Abdeljebbar)
Department of Multimedia, Signal and Communications System, National Institute of Posts and Telecommunications
Avenue Allal El Fassi, Madinat Al Irfane, Rabat, Morocco
(Email: abj.mourad@gmail.com)
(Received Apr. 18, 2017; revised and accepted July 30 & Sept. 2, 2017)

Abstract

Nowadays, the users' authentication is a main aspect of any security system in which network access should be completed by only accepted users. For this purpose, the EPS network uses EPS-AKA procedure to authenticate the mobile users. In this paper, we present and analyze this procedure and we propose a new solution in order to manage its weaknesses and vulnerabilities. This solution, called Improved EPS-AKA, is a combination between the simplicity of deployment, the full mutual authentication and the secured communication between all network entities. Finally, our solution is checked and validated by the AVISPA model.

Keywords: 4G Security; Authentication; AVISPA; EPS-AKA; LTE-SAE

1 Introduction

During the last three decades, many new mobile networks have been developed after the first generation. The EPS (Evolved Packet System) network is one of the fourth-generation networks, which was modeled to increase not only the user data rate and mobile communications security, but also the network reliability. The EPS system is the result of combination between Long Term Evolution network (LTE) and System Architecture Evolution network (SAE) in which the LTE plays the role of an access network and the SAE as a core network [11]. In fact, the LTE is an evolved UTRAN connected directly to mobile users via base stations called eNodeB, while the SAE is an IP-based network, which contains many network elements to connect mobile users with outside networks. The main elements of SAE network are as follows [2, 11]:

1) Home Subscriber Server (HSS): A main subscribers database, which contains authentication parameters and subscription information of mobile users;

2) Mobility Management Entity (MME): The control node which handles control functionalities, such as security functionalities;

3) Packet Data Network Gateway (PGW): A bridge between SAE network and external IP-based networks in order to manage mobile users' data;

4) Serving Gateway (SGW): A bridge between E-UTRAN and PGWs in order to manage mobile users' data and mobility.

Regarding mobile user authentication, the 3GPP group has chosen the EPS-AKA (Evolved Packet System - Authentication and Key Agreement) as a procedure of mobile users' authentication in the EPS networks [1]. However, the procedure does not provide a real secure authentication protocol [3]. Many researches have been carried out to address this weakness by improving or modifying totally or partially the initial procedure. Currently, the most proposed solutions are designed to improve the initial procedure as it will not make a considerable change to the network. Cao *et al.* in [10] have made a survey on the security weaknesses of the 4G LTE network and the proposed solutions in the literature. Indeed, the EPS-AKA authentication mechanism was found vulnerable to several kinds of passive and active attacks, breakthrough of the privacy, Denial of Service attacks and some IP attacks. However, many solutions have been proposed to cover these issues but still doest not provide a real secure full authentication procedure like in [4–7, 15, 17].

With a wide range of potential applications, Machine Type Communication (MTC) is gaining a tremendous interest among mobile network operators, equipment vendors and research bodies. Regarding mobile networks, the MTC becomes an important part of the 4G and 5G network infrastructure in which the communication is established between different devices sharing similar features and the core network [12, 14]. Traditionally, each device must authenticate separately to others, which increase the signaling and communication latency between the serving network and the home network. To cover this issue, Giustolisi *et al.* have proposed a group-based AKA to authenticate a group of devices [14]. Despite the solution protects the permanent identity of the mobile users in the air links as the mobile user never send his IMSI, this identity is always in danger on the wired links as the

transmission is done in plaintext format. As well, the network privacy is in danger as it is vulnerable to traffic redirection attacks.

With the rapid development of the smart-cards, the smart-card-based password authentication as a two-factor authentication mechanism (2FA) becomes the subject of many research in the last two decades in which hundreds of this type of schemes have been proposed [27–29]. By the way, the 2FA is a mechanism for confirming the identity claimed by a mobile user by utilizing a combination of two different factors, which means that the user who owns the smart card and its corresponding password is the one who can connect to the server [29]. Indeed, password protection is the main challenge of the 2FA mechanism as it can be leaked from a compromised server or from the user itself. For this reason, several schemes have been proposed to cover these issues like Yi *et al.*s in [32] where they have proposed to distribute the password files and the users' data into several servers to prevent the password leakage from a compromised server, while Yan *et al.* in [31] have proposed to use a trusted device by users to avoid the password leakage from users' side. Although significant efforts have been made to develop safe and effective 2FA schemes, a small effort has been made to design systematic systems to evaluate the 2FA schemes. Hence, Wang *et al.* have proposed a systematic framework that contains a practical adversary model as well as a well-refined criteria set, which was used to evaluate and analyze 67 proposed schemes in the literature. Thus, the proposed schemes do not satisfy truly the security goals of the 2FA mechanism. Therefore, a new scheme has been proposed [29].

The reminder of this paper is organized as follows. In Sections 2 and 3, we describe and analyze the EPS-AKA procedure, while in Sections 4 and 5 we present and analyze the new solution. In Section 6, we describe the AVISPA tool that is used to test the new solution and Section 7 present the tests results. Finally, we draw our conclusions and interpretations.

2 EPS-AKA Protocol

The EPS-AKA procedure is an authentication protocol defined by 3GPP group for mobile users' authentication when they access to EPS network via E-UTRAN. The procedure is an improvement of UMTS-AKA used in 3G network in order to have a strong authentication protocol. The authentication procedure as shown in Figure 1 is described as follows [13]:

1) UE ⟶ MME: The user equipment (UE) starts the authentication procedure by sending its permanent identity IMSI to the serving MME in an attach request message;

2) MME ⟶ HSS: In the authentication information request message, the serving MME forwards the received IMSI and its network identity SNID to the home HSS;

3) HSS ⟶ MME: Consequently, the HSS checks the IMSI and the SNID in order to generate an authentication vectors array when the verification is done successfully. In the authentication information response, the HSS shares these authentication vectors with the serving MME. In fact, each authentication vector contains a random number (RAND), a local master key (KASME), an expected response (XRES) and an authentication token (AUTN);

4) MME ⟶ UE: The serving MME stores then the received authentication vectors and selects one to answer to UE's request. In the authentication request message, the serving MME sends the value of RAND and AUTN to the UE. For future authentication, the serving MME will select an unused authentication vector from its database or from previous serving MME;

5) UE ⟶ MME: Consequently, the UE calculates its AUTN and compares it with the received one in order to authenticate the home HSS. After a successful authentication, the UE calculates the value of RES and then sends it to the serving MME;

6) MME: Finally, the serving MME compares the value of RES and XRES to authenticate the UE.

3 Security Analysis of EPS-AKA Protocol

Currently, the EPS-AKA protocol presents many threats and weaknesses, which can affect the privacy and secrecy of the network and mobile users. We explain and analyze below some threats and possible attacks [3]:

- IMSI leakage: In the attach request message, the UE sends its permanent identity (IMSI) to the serving MME in plaintext without confirming, in advance, the honesty of this MME. Therefore, an attacker who can catch and read this message can easily identify this UE and then affect his privacy;

- SNID leakage: According to 3GPP specifications [1], the serving network identity (SNID) is a combination of mobile country code (MCC) and mobile network code (MNC). In the authentication information request message, the serving MME sends its identity to the home HSS in plaintext. Therefore, an attacker who can catch and read this message can easily identify this MME and then impersonate the serving network;

- GUTI Tracking: The radio interface is a weak space against attacks. Therefore, the knowledge of the UE's temporary identity (GUTI) may affect the privacy of that user in which a fake eNodeB or MME

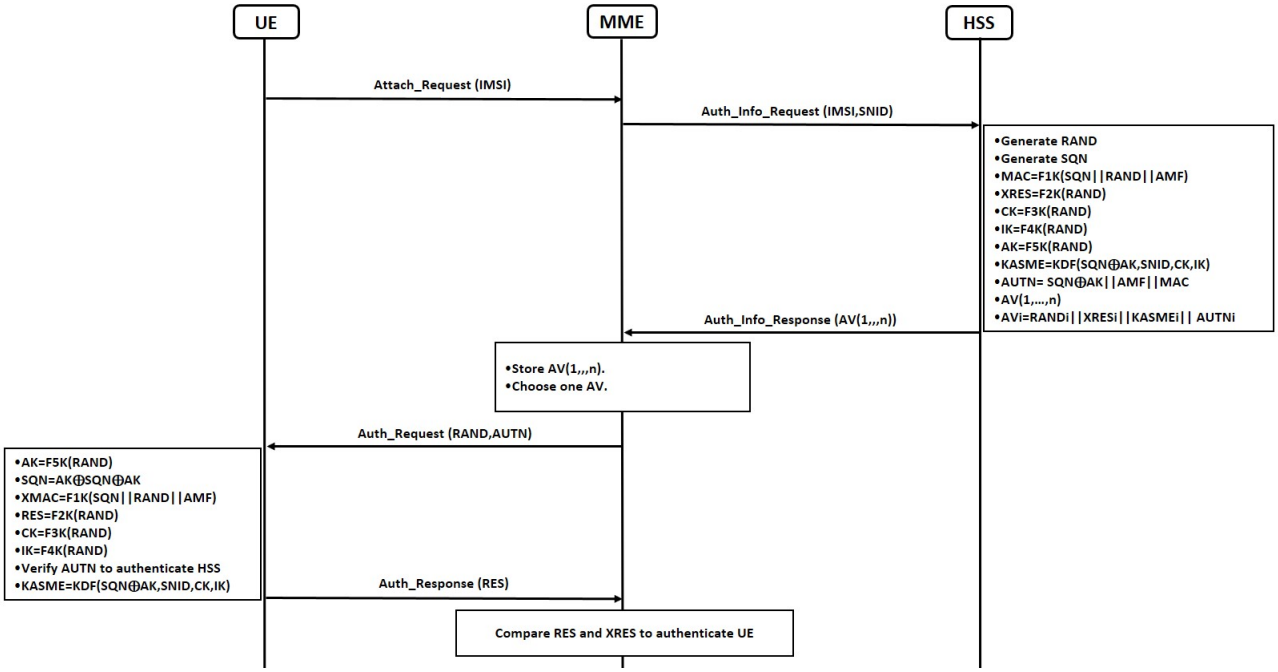


Figure 1: EPS-AKA procedure

can use this information to ask the UE to send its permanent identity and then affect the UE's privacy;

- Key leaked: Usually, the transmission over wired links is performed in plaintext, so an attacker can easily catch the authentication vectors shared with the serving MME and then gets the value of session keys. Therefore, the attacker can affect the secrecy of UE's communications;
- Link leakage: The transmission between EPC network entities is not protected against attacks because the transmission is performed in plaintext. Therefore, an attacker can easily catch the shared information and then affect the network privacy and secrecy;
- Traffic redirection: According to 3GPP specifications [1], the UE uses the value of AUTN to authenticate the HSS and the serving MME uses the value of RES to authenticate the UE, while the UE does not authenticate the MME and the MME does not authenticate the HSS. Therefore, the traffic redirection from an honest network to a fake network can be done by attackers.

4 Improved EPS-AKA Protocol

As described above, the EPS-AKA protocol still has some weaknesses related to the privacy and secrecy of the network and mobile users. To avoid such situation, the communication between UE and network in one side and between network elements in the other side should be protected. As well, the involved entities in the authentication procedure should be authenticated by each other's and the identity of the UE and network should be kept away from the third parties. Therefore, we propose the following changes to the original procedure:

- The communication between the involved entities in the authentication procedure is protected by the asymmetric cryptography system [20];
- The UE's permanent identity (IMSI) is hidden from the serving MME;
- The UE uses new symmetric key (NK) calculated by applying the XOR (exclusive or) operator to a generated random key (RK) and the original symmetric key (K) that is installed on its USIM card. The purpose of this new symmetric key is to protect the procedure from the replay attacks [18];
- The UE and HSS use a new secret parameter, called User Validation Parameter (UVP), to authenticate the UE by the HSS. The purpose of this new parameter is to protect the UE to be impersonated by an attacker when the UE's permanent identity is leaked;
- The serving MME and the home HSS use two new secret parameters, called Serving Network Validation Parameter (SVP) and Home Network Validation Parameter (HVP). The SVP is used to authenticate the serving MME by the home HSS while the HVP is used to authenticate the home HSS by the serving MME. Indeed, the two new parameters are an agreement between the home network and the serving network.

The improved EPS-AKA procedure is illustrated and explained below (Figure 2):

1) UE ⟶ MME: Firstly, the UE generates a random key RK to calculate a new symmetric key NK by applying the XOR operator to this random key and the original symmetric key K: $NK = RK \oplus K$. After that, it executes the hash function H on the new symmetric key and the UVP parameter: $UA = H(NK.UVP)$. By using the HSS's public key (Pkh), it encrypts the values of UA, RK and IMSI in which the result is sent to the serving MME in the attach request message with the value of the hybrid IMSI (HIMSI). In fact, the HIMSI is the first five digits of IMSI, which are related to country and network codes;

2) MME ⟶ HSS: The serving MME uses the HIMSI to get the HSS's public key from its database to encrypt its network identity (SNID), its SVP parameter and the encrypted value sent by the UE. Then, the result is sent to the home HSS;

3) HSS ⟶ MME: The HSS uses its private key (Prh) to decrypt the received messages and gets the values of UA, IMSI, RK, SNID and SVP. In order to authenticate the serving MME, the HSS uses the SNID to get the value of SVP parameter and then compares it with the received one. After a successful authentication, it uses the IMSI to get the value of the UVP parameter and the symmetric key K in which the latter is used to calculate the new symmetric key by the same way as UE: $XNK = RK \oplus K$. In order to authenticate the UE, it calculates the value of $XUA = H(XNK.UVP))$ and compares it with the value of UA. After a successful authentication of MME and UE, the HSS executes the hash function H on the SNID and the XNK: $MA = H(XNK.SNID))$ in which the result is encrypted by the UE's public key (Pku). Moreover, it generates an authentication vectors array, which is encrypted by the MME's public key (Pkm) with the UE's public key (Pku), the home network identity (HNID), the HVP parameter and the value of MA encrypted by the Pku key. Finally, the result is sent to the serving MME in the authentication response message;

4) MME ⟶ UE: The serving MME uses its private key (Prm) to decrypt the received message and then authenticates the HSS by verifying the HVP parameter. After a successful authentication of the HSS, it stores all data and choose one authentication vector (AV) to answer to UE's request. By the received UE's public key (Pku), it encrypts its identity SNID, its public key Pkm, the value of encrypted MA and the values of RAND and AUTN obtained from the selected vector, which are sent to UE in the authentication request message;

5) UE ⟶ MME: The UE decrypts the received messages by its private key (Pru). In order to authenticate the serving MME, the UE calculates the value of XMA by using the value of its new symmetric key NK and compares it with the received MA. After a successful authentication of the serving MME, it calculates the value of AUTN and compares it with the received one in order to authenticate the HSS. After a successful authentication of the serving MME and the home HSS, it calculates the value of RES and encrypts it with the received Pkm. Finally, the result is sent to the serving MME;

6) MME: Finally, the serving MME compares the values of RES and XRES to authenticate the UE.

5 Security Analysis of Improved EPS-AKA Protocol

The improved EPS-AKA protocol is designed to be a full mutual authentication protocol and solve the problem of the privacy and secrecy of the original protocol. A security analysis of this protocol is explained below according to the threats mentioned in Section 3:

- IMSI leakage: The UE encrypts its permanent identity (IMSI) by the HSS's public key before sending it to the serving MME. Therefore, the UE's identity is protected even if the serving MME is an attacker because it will need the HSS's private key to decrypt the message;

- SNID leakage: Using the SVP parameter by the HSS to authenticate the serving MME can protect the serving network from being impersonated by an attacker even if the SNID is disclosed. In addition, the transmission of this parameter is encrypted by the HSS's public key, which means that the attacker needs to know the HSS's private key to know the value of SVP parameter;

- GUTI Tracking: The transmission of the UE's permanent identity is protected by the HSS's public key. Therefore, an attacker cannot get the value of this identity even if he knows the value of his temporary identity, because he will need the HSS's private key to decrypt the message;

- Key leaked: The transmission of the UE's authentication vectors is protected by the MME's public key. Moreover, the serving MME must be authenticated successfully by the HSS before having these vectors. Therefore, a fake MME needs to be authenticated successfully by the HSS and needs to know the private key of the true MME to have these vectors and thus the session keys;

- Link leakage: The transmission of the authentication messages is protected by the asymmetric cryptogra-

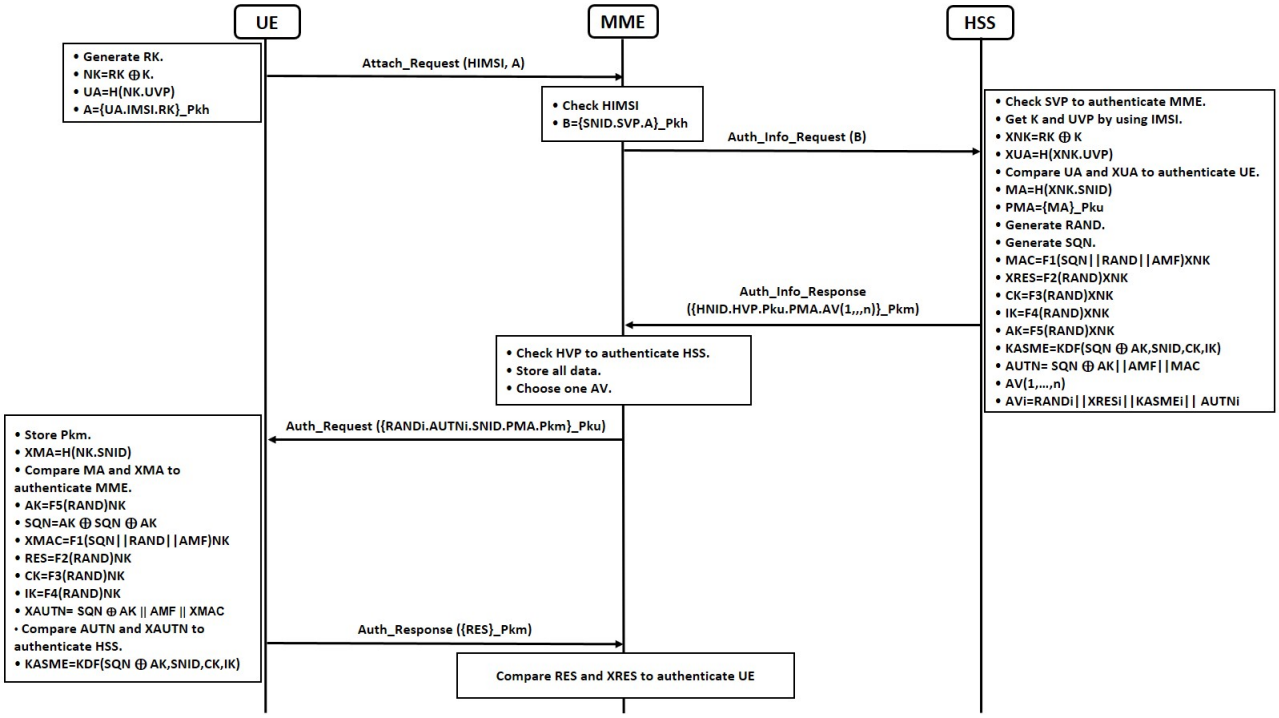


Figure 2: Improved EPS-AKA procedure

phy, which protects the network links against the disclosure of the shared information;

- Traffic redirection: The HSS needs to authenticate successfully the UE and the serving MME before sharing any information related to this UE's authentication. In addition, the serving MME needs to authenticate successfully this HSS before sending an authentication request message to the UE. Similarly, the UE needs to authenticate successfully the home HSS and the serving MME before confirming its access to the network in which the serving MME accepts this access if the authentication of that UE is succeeded. Therefore, the improved EPS-AKA is resistant against redirection attacks because each involved entity in the authentication procedure is authenticated by the others.

Furthermore, many schemes have recently been proposed to also secure the authentication procedure and solve the privacy problem in EPS networks. We present below three proposed schemes in order to compare them with our solution. The result of this comparison is shown in Table 1.

Hamandi *et al.* have proposed a privacy-enhanced to the initial EPS-AKA in which the aim is to minimize the transmission of IMSI and remove the linkability between the IMSI and GUTI [16]. In fact, the idea is to replace the permanent identity by a dynamic identity based on the RAND value and redefine the temporary identity by removing the MCC and MNC from the GUMMEI and calculating the M-TMSI by using the encryption and integrity keys of the NAS traffic. These keys are shared between the UE and the serving MME. In addition, the transmission of IMSI in the first attach is protected by the home HSS's public key and randomized by a timestamp. However, the transmission delay can affect the whole procedure if the request doest not arrive to the home HSS in the acceptable delay, hence the HSS will reject the request. This case is mostly happened when the serving MME and the home HSS are belongs to different networks (roaming case). As well, the transmission over the wired links is achieved in plaintext, which can affect the transferred information.

Wang *et al.* have proposed in [30] an improved privacy-preserving to Li *et al.* scheme [19] as it is found vulnerable to user anonymity violation attack and offline password guessing attack. Indeed, the both schemes are based on the use of a password-based smart card when a mobile user tries to access a foreign network and this password is used in the authentication parameters. However, the human action can affect the whole procedure as the human being can forgot the password of his or her smart-card or make multiple times a wrong password, which will block the smart-card and then the network access will not be possible. In addition, the human user can use the same password to access other applications and servers, such as e-mails, credit cards and so on, hence if the password is leaked from a non-secure access to these application and/or devices, the mobile user privacy can be affected.

Ramadan *et al.* have proposed in [22] a new scheme to solve the problem of radio attacks such as base station attacks in order to provide user-to-user mutual authentication and key agreement security in which the authentica-

Table 1: EPS network authentication schemes comparison result

	Our scheme	K. Hamandi scheme	D. Wang scheme	M. Ramadan scheme
IMSI leakage	**Safe**	**Safe**: The IMSI is encrypted before being sent.	**Safe**: The identity used instead of the IMSI is protected before being sent.	**Safe**: The hash value of the IMSI is used instead of the IMSI.
SNID leakage	**Safe**	**Unsafe**: The SNID is sent in plaintext.	**Unsafe**e: The identity (IDFA) used instead of the SNID is sent in plaintext.	**Safe**: The hash value of the SNID is used instead of the SNID.
GUTI Tracking	**Safe**	**Safe**: The new identity is calculated separately in both sides (UE and MME).	Not specified.	**Safe**: The hash value of the IMSI/GUTI is used instead of the IMSI/GUTI.
Key leaked	**Safe**	**Unsafe**: The AVs are sent in plaintext.	**Safe**: The session keys are calculated separately in both sides (foreign agent and mobile user).	**Safe**: The session keys are calculated separately in both sides (user A and user B).
Link leakage	**Safe**	**Unsafe**: The transmission over the wired links is done in plaintext.	**Safe**: A pre-shared symmetric key is used between foreign agent and home agent.	**Unsafe**: The transmission between HSS and MME is not secured.
Traffic redirection	**Safe**	**Unsafe**: The property of full mutual authentication is not applied.	**Unsafe**: The property of full mutual authentication is not applied.	**Unsafe**: The property of full mutual authentication is not applied.

tion is done between mobile users and the serving MME, while the HSS is used only to generate the system parameters from the security parameters. In fact, the proposed scheme is based on the Designated Verifier Proxy Signature (DVPS) in which the network is operate as a proxy and a non-trusted party. However, the idea of doing a user-to-user mutual authentication is interesting but the solution doest not provide any network privacy protection as the communication between the home network and the serving network is not discussed.

6 AVISPA Description and Architecture

The validation of security protocols is more difficult than normal communication protocols. Recently, many researches have been done to design validation tools by using the formal verification mechanism. R. Patel *et al.* in [21] have made a comparative study on the existing security verification tools in which they have concluded that the Scyther and AVISPA tools are the most efficient to verify and falsify security protocols. Therefore, we have chosen AVISPA to validate our solution.

The AVISPA project has been funded by European Community under the Information Society Technologies Program in which the High-Level Protocol Specification Language (HLSPL) was chosen as the programming language for the formulation of security protocols [23]. In fact, the HLPSL program is translated to intermediate format (IF) in order to be used by the four following back-ends tools: OFMC, CL-AtSe, SATMC and TA4SP (Figure 3) [26].

The HLPSL specification is defined by roles instead of messages, so it is sometimes difficult to confirm the correspondence between the HLPSL program and what the protocol designer wants to design. For this reason, an animator tool, called Security Protocol Animator (SPAN), has been developed to write, animate and understand the HLPSL specifications and build a Message Sequence Charts (MSC) of the protocol. In addition, the SPAN tool can simulate the possible attacks by constituting an active intruder [9].

7 Specification and Validation

The validation tests of the Improved EPS-AKA have been performed by the SPAN tool in which the security features to be simulated are mentioned in the HLPSL program goals section. By the way, our HLPSL program is divided into six sections in which the first three ones define the

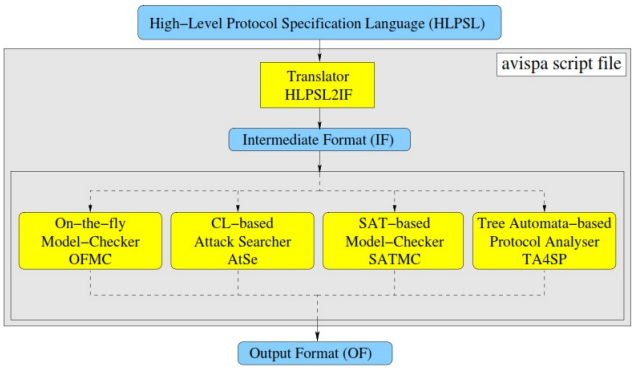


Figure 3: AVISPA Architecture

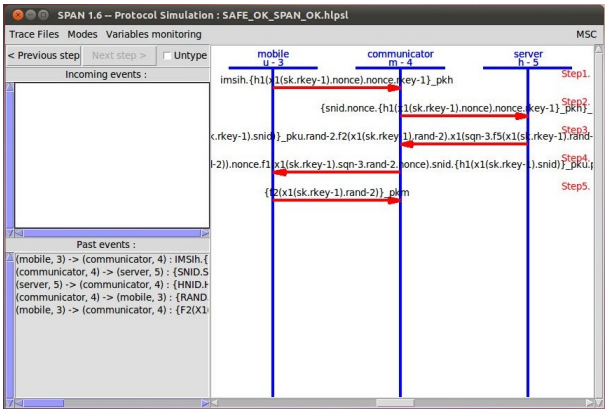


Figure 4: Improved EPS-AKA protocol simulation

role of UE, MME and HSS while the last three ones define the sessions, environment and goals of the protocol [24]. The improved EPS-AKA protocol simulation is illustrated in the Figure 4.

In order to check the robustness of the protocol, the following security features have been verified:

Secrecy: The expression secret(K,k,{A,B}) in the HLPSL program means that the value K produced or selected by a role played by the agent A is a shared secret between this agent and the agent B in which k is used to identify the secrecy goal in the goals section. In our case, the values of IMSI, K and UVP should be kept secret between the UE and the HSS while the values of SVP and HVP should be kept secret between the HSS and the MME;

Full mutual authentication: The full mutual authentication is obtained when the involved entities in the authentication procedure authenticate each other's. The verification of such goal is done by declaring the authentication request in the agent to be authenticated and the authentication witness in the authenticator. In our case, the UE authenticates the HSS by checking the AUTN value and the HSS authenticates the UE by checking the UA value. As well, the UE authenticates the MME by checking the MA value and the MME authenticates the UE by checking the RES value. Finally, the HSS authenticates the MME by verifying the SVP parameter and the MME authenticates the HSS by verifying the HVP parameter.

```
% OFMC
% Version of 2006/02/13
SUMMARY
  SAFE
DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
PROTOCOL
  /home/span/span/testsuite/results/SAFE_OK_SPAN_OK.if
GOAL
  as_specified
BACKEND
  OFMC
COMMENTS
STATISTICS
  parseTime: 0.00s
  searchTime: 0.02s
  visitedNodes: 1 nodes
  depth: 0 plies
```

Figure 5: Improved EPS-AKA simulation results by OFMC

```
SUMMARY
  SAFE

DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
  TYPED_MODEL

PROTOCOL
  /home/span/span/testsuite/results/SAFE_OK_SPAN_OK.if

GOAL
  As Specified

BACKEND
  CL-AtSe

STATISTICS

  Analysed   : 0 states
  Reachable  : 0 states
  Translation: 0.01 seconds
  Computation: 0.00 seconds
```

Figure 6: Improved EPS-AKA simulation results by Cl-AtSe

In addition, the protocol robustness is also simulated by introducing in the intruder knowledge any information that can be known by the attackers. In fact, the Figures 5 and 6 show the results of the simulation by OFMC and Cl-AtSe verification tools [8,25]. As we can see, the summary of the simulation shows that the protocol is safe and that no attack has been detected. Therefore, all security goals mentioned above have been satisfied.

8 Conclusions

The EPS-AKA protocol objectives are quite similar to those for UMTS-AKA, used in 3G networks. The enhancement of EPS-AKA is that it provides implicit serving network authentication, which is achieved by binding an appropriate key, KASME, to the serving network identity. However, the existing EPS-AKA protocol does not offer a real secure authentication. Therefore, in this paper we proposed a new method called Improved EPS-AKA, which offers a secure authentication procedure by protecting any message exchange by the asymmetric cryptography system. Moreover, it protects the confidentiality of the UE and network even if the serving MME and the HSS are belong to the same network. The AVISPA tool was used to validate this new solution in which the result shows that it is resistant against attacks such as a replay attacks. The choice of AVISPA tool to validate our solution is due to the validation tools that it contains, which are used to validate the security protocols such as the authentication protocols. However, the security goals defined in a formal model may be different from the others, hence the simulation results may be different also. In addition, new attacks may be appeared in the future and may be difficult to model on a formal verification model. Therefore, if the security model used is currently the right one, the correctness of a security proof generally depends on the attacking experience.

References

[1] 3rd Generation Partnership Project (3GPP), *3rd generation partnership project; technical specification group services and system aspects: 3GPP system architecture evolution (SAE)/security architecture*, Technical Report 3GPP TS 33.401, V15.0.0, June 2017.

[2] 3rd Generation Partnership Project (3GPP), *3rd generation partnership project; technical specification group services and system aspects: Network architecture*, Technical Report 3GPP TS 23.002, V14.1.0, Mar. 2017.

[3] M. Abdeljebbar and R. E. Kouch, "Security analysis of LTE/sae networks over e-utran," in *Proceedings of The Second International Conference on Information Technology for Organizations Development (IT4OD'16)*, Fez, Morocco, 2016.

[4] J. B. Abdo, J. Demerjian, H. Chaouchi and G. Pujolle, "Ec-aka2 a revolutionary aka protocol," in *Proceedings of The International Conference on Computer Applications Technology (ICCAT'13)*, Sousse, Tunisia, Jan. 2013.

[5] M. A. Abdrabou, A. D. E. Elbayoumy and E. A. El-Wanis, "LTE authentication protocol (EPS-AKA) weaknesses solution," in *Proceedings of The IEEE Seventh International Conference on Intelligent Computing and Information Systems (ICICIS'15)*, Cairo, Egypt, Dec. 2015.

[6] K. A. Alezabi, F. Hashim, S. J. Hashim and B. M. Ali, "An efficient authentication and key agreement protocol for 4g (LTE) networks," in *Proceedings of The IEEE Region 10 Symposium*, Kuala Lumpur, Malaysia, Apr. 2014.

[7] C. G. Apostol and C. Racuciu, "Improving LTE EPS-AKA using the security request vector," in *Proceedings of The Seventh International Conference on Electronics, Computers and Artificial Intelligence (ECAI'15)*, Bucharest, Romania, June 2015.

[8] D. Basin, S. Modersheim and L. Vigano, "OFMC: A symbolic model-checker for security protocols," *International Journal of Information Security*, vol. 4, no. 3, pp. 181–208, 2005.

[9] Y. Boichut, T. Genet, Y. Glouche and O. Heen, "Using animation to improve formal specifications of security protocols," in *Proceedings of the Second National Conference on Security in Network Architectures and Information Systems*, Annecy, France, 2007.

[10] J. Cao, M. Ma, H.Li, Y. Zhang and Z. Luo, "A survey on security aspects for LTE and LTE-A networks," *IEEE Communications Surveys and Tutorials*, vol. 16, no. 1, pp. 283–302, 2014.

[11] C. Cox, *An introduction to LTE, LTE-advanced, SAE and 4G mobile communications (1st, in English)*, pp. 352, 2012.

[12] O. Dementev, "Machine-type communications as part of LTE-advanced technology in beyond-4g networks," in *Proceedings of The Fourteenth Conference of Open Innovations Association (FRUCT'13)*, Espoo, Finland, Nov. 2013.

[13] D. Forsberg, G. Horn, W. D. Moeller and V. Niemi, *LTE Security (2ed, in English)*, pp. 368, 2013.

[14] R. Giustolisi, C. Gehrmann, M. Ahlstrm and S. Holmberg, "A secure group-based AKA protocol for machine-type communications," in *Proceedings of The International Conference on Information Security and Cryptology (ICISC'16)*, Seoul, South Korea, 2016.

[15] Z. J. Haddad, S. Taha and I. A. S. Ismail, "SEPS-AKA: A secure evolved packet system authentication and key agreement scheme for LTE-A networks," in *Proceedings of The Sixth International Conference on Wireless and Mobile Networks (WiMONe'14)*, Sydney, Australia, Dec. 2014.

[16] K. Hamandi, J. B. Abdo, I. H. Elhajj, A. Kayssi and A. Chehab, "A privacy-enhanced computationally-efficient and comprehensive LTE-AKA," *Computer Communications*, vol. 98, pp. 20–30, 2017.

[17] K. Hamandi, I. Sarji, A. Chehab, I.H. Elhajj and A. Kayssi, "Privacy enhanced and computationally efficient hsk-AKA LTE scheme," in *Proceedings of The Twenty-seventh International Conference on Advanced Information Networking and Applications Workshops (WAINA'13)*, Barcelona, Spain, Mar. 2013.

[18] A. Jesudoss and N. P. Subramaniam, "A survey on authentication attacks and countermeasures in a distributed environment," *The Indian Journal of Computer Science and Engineering (IJCSE'14)*, vol. 5, no. 2, pp. 71–77, 2014.

[19] H. Li, Y. Yang and L. Pang, "An efcient authentication protocol with user anonymity for mobile networks," in *Proceedings of The Wireless Communications and Networking Conference (WCNC'13)*, Shanghai, China, Apr. 2013.

[20] S. Nithya and D. E. G. D. P. Raj, "Survey on asymmetric key cryptography algorithms," *Journal of Advanced Computing and Communication Technologies (JACOTECH'14)*, vol. 2, no. 1, pp. 1–4, 2014.

[21] R. Patel, B. Borisaniya, A. Patel, D. Patel, M. Rajarajan and A. Zisman, "Comparative analysis of formal model checking tools for security protocol verification," in *Proceedings of The International Conference on Network Security and Applications (CNSA'10)*, Chennai, India, July 2010.

[22] M. Ramadan, F. Li, C.X. Xu, A. Mohamed, H. Abdalla and A. Abdalla, "User-to-user mutual authentication and key agreement scheme for LTE cellular system," *International Journal of Network Security*, vol. 18, no. 4, pp. 769–781, 2016.

[23] The AVISPA Team, *Avispa v1.1 user manual*, Technical Report www.avispa-project.org, June 2006.

[24] The AVISPA Team. *A beginners guide to modelling and analyzing internet security protocols*, Technical Report www.avispa-project.org, June 2006.

[25] M. Turuani, "The cl-atse protocol analyser," in *Proceedings of The Seventeenth International Conference on Term Rewriting and Applications (RTA'06)*, Seattle, USA, Aug. 2006.

[26] L. Vigan, "Automated security protocol analysis with the avispa tool," *Electronic Notes in Theoretical Computer Science*, vol. 155, pp. 61–86, 2006.

[27] D. Wang, Q. Gu, H. Cheng and P. Wang, "The request for better measurement: A comparative evaluation of two-factor authentication schemes," in *Proceedings of The Eleventh ACM on Asia Conference on Computer and Communications Security (ASIA CCS'16)*, Xi'an, China, 2016.

[28] D. Wang and P. Wang, "Understanding security failures of two-factor authentication schemes for real-time applications in hierarchical wireless sensor networks," *Ad Hoc Networks*, vol. 20, pp. 1–15, 2014.

[29] D. Wang and P. Wang, "Two birds with one stone: Two-factor authentication with security beyond conventional bound," *IEEE Transactions on Dependable and Secure Computing*, vol. 99, no. 99, pp. 1–1, 2016.

[30] D. Wang, P. wang and J. Liu, "Improved privacy-preserving authentication scheme for roaming service in mobile networks," in *Proceedings of The Wireless Communications and Networking Conference (WCNC'14)*, Istanbul, Turkey, Apr. 2014.

[31] Q. Yan, J. Han, Y. Li and R. H. Deng, "On limitations of designing leakage-resilient password systems: Attacks, principles and usability," in *Proceedings of The Nineteenth Network and Distributed System Security Symposium (NDSS'12)*, San Diego, California, Feb. 2012.

[32] X. Yi, S. Ling and H. Wang, "Efficient two-server password-only authenticated key exchange," *IEEE Transactions on Parallel and Distributed Systems*, vol. 24, no. 9, pp. 1773–1782, 2013.

Biography

Mourad Abdeljebbar is a doctoral student at the National Institute of Posts and Telecommunications (Rabat, Morocco), where he graduated as a telecommunication engineer in 2008. After that, he joined Nokia Networks Morocco as a R4 circuit switching core consultant where he worked on several projects related to 3G and 4G networks. His research interests include authentication protocols, mobile networks, network security, cryptography and telecommunications networks.

Rachid El Kouch is a professor at the National Post and Telecommunications Institute (Rabat, Morocco). Since 1981, he is responsible for the PABX laboratory attached to the Systems and Communications Department at the INPT. Pr. EL KOUCH began his career as a telecommunications assistant engineer. He worked as an assistant in the INPT's electricity and electronics laboratory. In 1989, he graduated from the University of Colorado with a Master of Science in Telecommunications degree at Boulder, USA. In 2005, he received his PhD in Applied Mathematics from Mohammed Ben Abdellah University - Faculty of Science and Technology of Fez-USMBA. Between 1990 and 1992, he was a specialized tutor for several technical inspectors. It was a technology transfer program under a convention between France and Morocco. He supervised several internships for engineering students in the 2nd year of the engineering cycle and over 100 graduation projects for engineering students at the end of their training at INPT. He has been Deputy Director of Internships and Relationships with Enterprises since 2008. In 2010, he was asked for the position of Deputy Director of Continuing Education. In addition, he is a member of the research team Phare (UFR MDA) of the Laboratory of computer science and mathematics of the USMBA. Since December 2005, when the training project was launched (collaboration between UBC and Lyon1 USMBA), he supervised several modules on the electronic platform Miage (Lyon 1): B202- Bases of telecommunications and B214 - Network protocols. He has more than 30 articles in conferences and national / international journals on telecommunications, networks and security. He has also been a reporter for more than twenty theses (Ph.D).

Benchmark Datasets for Network Intrusion Detection: A Review

Yasir Hamid[1], V. R. Balasaraswathi[1], Ludovic Journaux [2] and M. Sugumaran[1]
(Corresponding author: Yasir Hamid)
Department of Computer Science and Engineering, Pondicherry Engineering College[1]
East Coast Road, Pillaichavadi, Puducherry 605014, India
Lab. LE2I UMR, CNRS 6306 AgroSup, F-21000, Dijon, France[2]
(Email: bhatyasirhamid@email.address)
(Received Feb. 27, 2017; revised and accepted June 25, 2017)

Abstract

Network Intrusion Detection is the process of monitoring the events occurring in a computer system or the network and analyzing them for the signs of possible intrusions. An intrusion is a potentially harmful activity of malicious user, aimed at compromising the confidentiality, availability and integrity of the system. Over the decades intrusion detection (ID) problem has been visited by the researchers in various available environments like finite state automata, rule based systems, Markov probabilistic approach, statically sought solutions and most popular of all data mining and machine learning techniques. The prerequisite for data mining is that data should be present and there should be some hidden patterns in the data which need to be unearthed. In this work, we intend to provide a thorough review of the benchmark datasets available for Network Intrusion Detection (NID) which researchers in the field can use to train and test their models. In addition, this work as the first of its kind implements k-NN a simple most instance based type of classifier over all the datasets that doesn't require a well planed and monolithic training phase, across different neighborhood sizes. Results show that off all the datasets k-NN performs better on NSL-KDD dataset due to the fact that NSL-KDD doesn't have any redundant network connections and connections being fairly distributed across all the classes.

Keywords: Benchmark Data-sets; Classification; IDS; k-NN; Machine Learning; Network Security

1 Introduction

Due to the popularity of Internet in various important institutions of life like health, business, education and military it has been under the continuous threat from the people with dangerous mindset [24]. Hence, the need of securing the networks from such people is more than ever before. Over the years a boundless of methods, tools and devices have surfaced to secure the networks and computer systems ranging from anti viruses, firewalls, vulnerability testing and elimination of the programming errors that provide an back door entry point to the system [3, 20, 21].

All these devices for network security can be categorized as reactive or proactive [14], where proactive methods are based on the assumption of that security can be guaranteed and hence it is absolutely possible. Since Internet is distributed in nature and doesn't have any central security component. With the landscape of potentially harmful tools growing with each day and the fact that hackers are getting more smarter than ever before, prevention is no more effective. No matter how more securely a network is laid down, hackers will eventually find a way to break into the system and hence threaten the confidentiality, availability and integrity of the system.

This leads us to the reactive system that make use of network and system logs to unearth the foot prints of possibly disastrous network connections, one such devices that has got popularity and attracted tremendous research is Intrusion Detection System (IDS. Operating on a series of steps an IDS starts with collection of data from the networks, followed by data manipulation and finally a test for abnormality, that classifies a network connection as normal or abnormal. In the earlier phases, the research focal point lied with rule-based expert methods and statistical procedures. However these rule based systems tend to be less effective on the larger datasets and mostly end up performing their worst. Along these lines, a considerable measure of machine learning (ML) techniques have been acquainted to tackle the problem of ID. Both supervised and unsupervised ML techniques have been equally popular in IDS. IDS comes handy not only in successful detection of intrusions but is also effective in monitoring attempts to break security, which is indispensable for timely countermeasures against the ongoing intrusion activity [25].

An IDS can be categorized as Host based Intrusion Detection Systems (HIDS) or Network based Intrusion Detection Systems (NIDS) determined by its position of placement over the network system [23]. HIDS uses the data stored on single host and compromises of an agent on host that is entrusted with identifying the intrusions by examining the system calls, application logs, file modifications *etc.*

The advantage of HIDS is its simple nature and ease of installation, but the problem is that they are not immune against distributed and coordinated attacks. Contrary to this, NIDS collects and analyses the traffic communicated over the network. So it can carry out intrusion detection with security measures reflecting entire network information, hence preventing the attacks from reaching the hosts [1]. In view of the recognition technique at the most fundamental level an IDS can be classified as misuse based or anomaly based. Misuse based detection is trained on the labeled set compsinsing of instances labeled as as normal or an attack. The algorithm can detect the known attacks with high confidence but fail to detect novel attacks or for that case the simple variations of the known attacks [9]. Contrary to this anomaly based approaches are trained on the unlabelled data and in the training phase they build up the model for the normal connections. Over the span of operation the system captures the data and compares it with the model for the normal data. If it differs by more than some threshold then it is classified as an attack otherwise a legitimate connection. The power of anomaly based system comes with the ability of the detecting novel attacks, but this comes with the higher false alarm rates.

For the last few decades researchers have approached the ID problem in various models and most popular of the all is the application of data mining methods [13]. Data mining explores and analyzes gigantic datasets to unearth the useful and understandable patterns from the data. A prerequisite for the data mining is the availability of lots of data, presence of patterns in the data and presence of no simple model to understand the data. As for network intrusion detection is considered many datasets have surfaced over the years, with varied applicability, attack ranges and various capturing methods. In this paper we attempt to document most of the datasets available for intrusion detection and in addition to that we try to classify all the datasets using k-NN. The objective of application of k-NN across datasets over different neighborhoods is just to get an insight how well the data is classified.

The remainder of the paper is organized as follows. Section 2 provides a brief review of the literature for network intrusion detection. A brief discussion of machine learning techniques is given in Section 3, Section 4 provides a detailed discussion of various datasets available, the results of k-NN over different neighborhoods across all the datasets is given in Section 5, finally the paper concludes in Section 6.

2 Literature Review

Machine Learning (ML) techniques have been well sought and highly popular for Network Intrusion Detection. A group of varied and widespread ML techniques spanning both supervised as well as unsupervised ML groups have been applied for network intrusion detection. As for supervised ML techniques are considered almost all the techniques have been equally popular, like Decision Trees [12] Artificial Neural Networks (ANN) [16], Support Vector Machines [10] and for unsupervised machine learning techniques k-NN [15], have been pretty popular. Moreover, there have a lot been of efforts in designing both hybrid [22] and ensemble [19] models for the network intrusion detection. Review of the literature enlightens about the fact that KDD99 has been well accepted and thoroughly used dataset among the community. Lately, Bio-inspired algorithms also have been applied in network intrusion detection mostly in pre-processing to select an optimal subset of features possibly non-redundant and highly correlated with the class and least correlation with other features Ant Colony Optimization [11], Cuttle Fish Algorithm [8], Particle Swarm Optimization [6]. Due to excessive computational requirements and inherent drawbacks of the dataset the full dataset has been used very seldom. So, mostly the researchers have relied on selecting a portion of the dataset and trained and tested the models on the extracted subset of the data.

3 A Review of Machine Learning Techniques

Machine Learning when considered in generic means to form meaningful predictive models from the tsunami of data. Its various prospects extend to anomaly detection, prevention of fraud, intrusion in networks, lifetime support such as DNA analysis, tumor detection, sentiment analysis in social networks, detection of objects in satellite imageries and making appropriate decisions in the automated vehicles. Machine learning actually boots in the training data, then learns from the samples and builds up a congruous model, next tests itself and then predicts the data or engenders decision from the live environment without being precisely programmed for each action [4].

Machine learning is segregated into three broad groups *i.e*, supervised learning, unsupervised learning and reinforcement learning.

3.1 Supervised Learning

Supervised learning confides over the training samples to build a prognosticative model. Supervised learning model employs over pair which encompasses the input object and desired output value [2]. There are many supervised learning techniques embracing Decision Tree learning, Support Vector Machine, K-nearest Neighbor, Naive Bayes and Random Forest. The decision tree learning

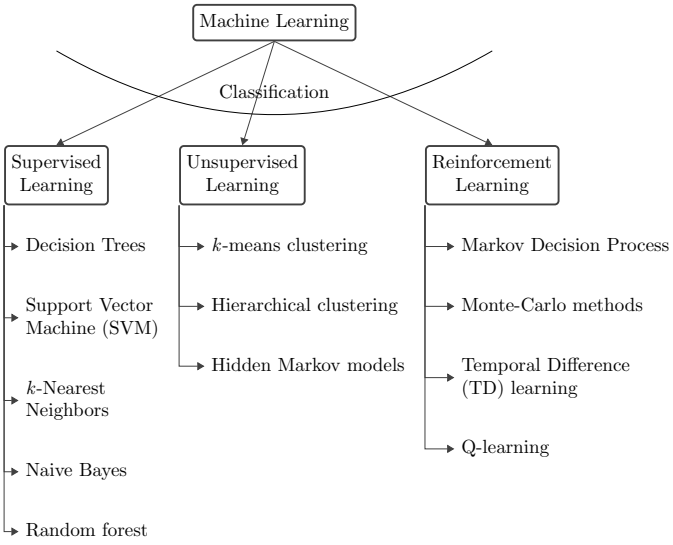


Figure 1: Machine learning classification

sets upon to create a prototype which will forecast the value of an objective variable by manipulating over a set of decision variables down the tree. Scrutinizing diverse decision variable along the path by traversing from the root node to leaf node which contains the value of the objective node. Support vector machine [5] endeavors to find out the perfect separating hyperplane that can perfectly classify the objects. Its ambition is to maximize the distance between the data points and minimize the distance between hyperplane and data points. The k-nearest neighbor is lazy non-parametric learning technique. The k-NN in contrast to SVM takes the entire training data for consideration where SVM disposes of the non-support vectors. Naive Bayes is a probabilistic classifier, where it weighs the probability of each event based on certain conditions associated with it. Random forest is aggregate of decision trees, with each tree of the forest promoting a vote over the decision made and finally aftermath will be label having a maximum number of votes.

Algorithm 1 Supervised Learning Technique

1: **procedure** SUPERVISED LEARNING
2: Fetch in training data set $X = \{\alpha^n, \beta^n\}$ α is data point; β is labelling
3: Build Predictive model based on the training set
4: Fetch Objective variable $new\alpha$ which needs to be labelled based on $new\alpha = \begin{cases} \epsilon^1, & \text{Class A} \\ \epsilon^2, & \text{Class B} \\ .... & \\ \epsilon^n & \text{Class N} \end{cases}$ Each ϵ is of different threshold values
5: **end procedure**

3.2 Unsupervised Learning

Unsupervised learning is enforced over data samples where there doesn't exist prior labeling of data samples. But Unsupervised Learning is still a conundrum, clear understanding will revolutionize this field. It produces a function to classify the data by hunting out the concealed structure within the data. It includes K-means clustering, Hierarchical Clustering and Hidden Markov model. K-means clustering works over the principle of feature congruence. Hierarchical clustering fashions up to form clusters building up either in top-down or bottom-up trend namely divisive and agglomerative. Hidden Markov model plays hardball over data with time series. Its principle is over probability based Bayesian network.

Algorithm 2 Unsupervised Learning Technique

1: **procedure** UNSUPERVISED LEARNING
2: Fetch data set $X = \{\alpha^1, \alpha^2, \alpha^3 .....\alpha^n\}$ where α represents a data point
3: Build Clustering model to form clusters
4: $\begin{cases} \text{if } \epsilon^1, & \text{then Group A} \\ \text{if } \epsilon^2, & \text{then Group B} \\ & \text{Using some distance measure} \\ .... & \\ \text{if } \epsilon^n, & \text{then Group N} \end{cases}$
form ϵ^n threshold values, thus forming N Clusters
5: Fetch Objective variable $new\alpha$ which needs to be classified $new\alpha = \begin{cases} \epsilon^1, & \text{Group A} \\ \epsilon^2, & \text{Group B} \\ .... & \\ \epsilon^n & \text{Group N} \end{cases}$
6: **end procedure**

3.3 Reinforcement Learning

Reinforcement Learning is primarily used by software agents which need to make a decision in an environment. This learning doesn't bother about immediate rewards, rather its ultimate motive is a win over the environment with high reward tardily [17]. Markov decision process, Monte Carlo methods, Temporal Difference learning and Q-learning are some of the reinforcement learning. The Markov decision process is used in footing where the fallouts are only partially in control. Monte Carlo methods equate over averaging the returns of the sample. Temporal Difference Learning is a perfect blending of Monte Carlo and Dynamic programming. Q-Learning works by recommending best selection policy for each action by analyzing the play by play over the past period of time.

3.4 *k*-NN

k-NN is an intelligible method which performs preposterously well. k-NN is dexterous in nature and used across various domains. It is actually a lazy learning algorithm.

Algorithm 3 Reinforcement Learning Technique

procedure Reinforcement Learning
  Fetch states γ in the Environment ξ
  for all γ_i in γ **do**:
    Find Best δ_i from δ assign reward value ω_i to it
  end for
where δ is the action and ω is the reward
  Given $new\gamma$ find action δ
  if δ_i have greater ω **then**:
    $new\gamma$ is assigned δ_i
  end if
  Ultimate goal to Win the ξ with best ω
end procedure

The k-factor gives the value of how many closest data points to be considered to conclude the classification of the observatory data point. In the point of fact, the k-factor is the deciding parameter of how well the algorithm can perform efficiently. Once the k-factor's value is decided, the voting among the k closest points are taken into consideration and label which has the maximum votes wins the classification. If the k-factor is determined is one, initially, it will have zero error in classification but the error curve will drop down to zero and then maneuver an upward movement indicating the increase in error rate, when k-factor is taken large, then happens under-fitting dispute. So it is indispensable to decide the impeccable value of the k-factor.

Algorithm 4 k-NN

procedure k-NN Algorithm
  Fetch in training data set $X = \{\alpha^n, \beta^n\}$ α is data point; β is labelling
  Determine the Value of K-factor
  Fetch Objective variable $new\alpha$ which needs to be labelled
  Based on K-Factor δ^n fetch the nearby data-points of the $new\alpha$ based on Distance measure ω
  for all i in $Countof\delta$ **do**:
    if ω have lesser ξ **then**:
      $\delta[] = \alpha_i$
    end if
  end for
Each ξ is of threshold distance measure
  Find the votes ζ_i $\delta_i = \begin{cases} \epsilon^1, & \text{Class A} \\ \epsilon^2, & \text{Class B} \\ .... & \\ \epsilon^n & \text{Class N} \end{cases}$
  Find β label which has maximum no of votes in ζ_i and this is the Final labelling to the $new\alpha$
end procedure

4 Data Sets

In the next few subsections, we provide an overview of various datasets used for network intrusions over the years. Although this list is by no ways a complete documentations of all the datasets, as users over the course of time have used randomly drawn subsets of the full dataset. We provide a review of the stable and very well known datasets, that are pretty popular in the ML fraternity.

4.1 DARPA98

DARPA98 was the first standard corpora for the assessment of network intrusion detection, collected and distributed by MIT Lincoln under the joint sponsorship of DARPA and ARFL. Over the decades the dataset has been popularly used for training and testing models of IDS. These assessments contributed fundamentally to the intrusion detection by giving guidance for research endeavors and a target adjustment of the specialized cutting edge. The dataset was formed by recording whole network traffic including the payload in Tcpdump format for each connection. Both real and simulated machines were used for the data collection and in general, the data collected was in the form of sniffed network traffic, Solaris BSM audit data, Windows NT audit data (in the case of DARPA 1999), file system snapshots aimed at identifying the intrusions that were being carried out against a test network while the data was being collected. The dataset is comprised of 7 weeks data for the training and 2 weeks for the testing purposes. More than three hundred instances of 38 attacks broadly categorized in 4 different groups like DOS, U2R, R2L and Probe are present in the dataset.

4.2 KDD99

The Third International Knowledge Discovery and Data Mining Tools Competition was held in conjunction with KDD99, The Fifth International Conference on Knowledge Discovery and Data Mining. The competition task was to build a network intrusion detector, a predictive model capable of distinguishing between "bad" connections, called intrusions or attacks and "good" normal connections. This database contains a standard set of data to be audited, which includes a wide variety of intrusions simulated in a military network environment and the dataset that was used was processed DARPA98 data and became popular as a KDD99 dataset. The KDD99 dataset is the most popular dataset among machine learning practitioners for network intrusion detection. Each connection record consists of 41 data fields and there is 42nd attribute designating the class to which the connection belongs. In total there are 23 attack groups in the data and there is also a class for normal connections. The 23 attack groups are broadly categorized into four attack families i.e, Dos, Probe, R2L and U2R. Each connection record is mixed in nature, with some attributes

Table 1: KDD attributes and their types

BASIC			CONTENT			TRAFFIC			HOST		
SNO	FEATURE	TYPE	SNO	FEATURE	TYPE	SNO	FEATURE	TYPE	SNO	FEATURE	TYPE
1	duration	C	10	hot	C	23	count	C	32	dst_host_count	C
2	protocol_type	N	11	num_failed_logins	C	24	serror_rate	C	33	dst_host_srv_count	C
3	service	N	12	logged_in	N	25	rerror_rate	C	34	dst_host_same_srv_rate	C
4	src_bytes	C	13	num_compromised	C	26	same_srv_rate	C	35	dst_host_diff_srv_rate	C
5	dst_bytes	C	14	root_shell	N	27	diff_srv_rate	C	36	dst_host_same_src_port_rate	C
6	flag	N	15	su_attempted	N	28	srv_count	C	37	dst_host_srv_diff_host_rate	C
7	land	N	16	num_root	C	29	srv_serror_rate	C	38	dst_host_serror_rate	C
8	wrong_fragment	C	17	num_file_creations	C	30	srv_rerror_rate	C	39	dst_host_srv_serror_rate	C
9	urgent	C	18	num_shells	C	31	srv_diff_host_rate	C	40	dst_host_rerror_rate	C
			19	num_access_files	C				41	dst_host_srv_rerror_rate	C
			20	num_outbound_cmds	C				42	class	
			21	is_hot_login	N						
			22	is_guest_login	N						

being nominal and others being numeric. The distribution of connections across different groups is uneven and there are more no of connections pertaining to two dominating attack groups i.e, Probe and R2L and also the normal connections. There are only a few instances of the R2L and U2R attacks. A large number of variations of KDD99 datasets some being the corrections of the base dataset while others being the carefully drawn subsets have surfaced up over the years. These 41 attributes are categorized into four broad groups as

- Basic features: attributes representing individual connections.
- Content features: The features within a connection as suggested by domain knowledge.
- Traffic features: features comprised of 2 second time window.
- Host features: attributes for assessing attacks lasting for more than 2 seconds.

Some of the attributes are nominal and most are continuous. To provide a better picture of the dataset, Table 1 records and groups the attributes into different categories, moreover for each attribute, we mention its nature, whether it is nominal or numeric.

In the following few subsections, we will provide a brief discussion of variations of KDD99 datasets.

4.2.1 Full KDD

Full KDD in total consists of 48, 98, 430 instances, the file is of 750mb. All the connections of the dataset fall under four groups. The frequency of the attacks is not normal, where there are few dominating classes like Neptune and smurf with 10, 72, 017 and 28, 07, 886 instances respectively. But there are also less frequent attack groups like Spy, Perl, PHP. This uneven distribution of connections across different attack groups hampers the detection rate of any classification system and results in a biased classifier.

4.2.2 Corrected KDD

The problem with the full KDD99 dataset is the massive repetitions of few instances. This hampers the effective classification as the results of most of the classifier are biased towards the dominant classes. Just to provide the researchers a potentially fair dataset to train and test model a variation of the full dataset called corrected KDD99 dataset was formed. A total of 3, 11, 029 network connections are present in corrected KDD dataset. Care was taken to maintain the representation of all the attacks and at the same time reduces the redundancy and repetitions.

4.2.3 10%KDD

Due to the enormous size of KDD dataset, it is practically not possible to use it fully for the training set and hence usually researchers select a subset of the dataset from the full set to train and test their models. One such carefully drawn a subset of the total subset has been pretty popular in the machine learning fraternity and is known as 10%KDD. In total, this fraction of the dataset has 4, 89, 843 instances. This compact size of 10%KDD makes it suitable for training and testing of IDS models. In total 4, 89, 843 connections are present in the dataset.

4.2.4 NSL KDD

NSL-KDD data set is a refined version of its predecessor KDD99 dataset formulated with the aim of minimizing the bias of the classifier. The advantage of NSL KDD dataset over full KDD are unbiased classification and reliable results as there are no redundant records in the dataset. Better detection rate offered by eliminating the duplicate results. A sort of balance and representation across the groups of attacks by selecting the number of instances from each group inversely proportional to original KDD dataset.

Table 2 given below provides a thorough analysis of all the variations of the KDD dataset present in literature. For each dataset, we present the number of instances of different attack classes. Of all the variations of the dataset corrected KDD99 data has highest a number of attacks, There are in total 38 attacks in the corrected KDD99 dataset, 23 of them are same as that of the other variations with 15 new attacks being incorporated.

Table 2: Analysis of variations of dataset

Attack	Total Instances						Attack Group
	Full KDD	Corrected	10%KDD	NSLKDD	NEW_ATTACKS	#Count	
back	2203	1098	2203	956	apache2	794	DOS
land	21	9	21	18	processtable	759	
neptune	1072017	58001	107201	41214			
pod	264	87	264	201			
smurf	2807886	164901	280790	2646			
teardrop	979	12	979	892			
satan	15892	1633	1589	3633	saint	736	PROBE
ipsweep	12481	306	1247	3599	snmpguess	2406	
nmap	2316	84	231	1493	mscan	1053	
portsweep	10413	354	1040	2931			
normal	972781	60593	97277	67343			NORMAL
guess_passwd	53	4367	53	53	worm	2	R2L
ftp_write	8	3	8	8	xlock	9	
imap	12	1	12	11	xsnoop	4	
phf	4	2	4	4	xterm	13	
multihop	7	18	7	7	httptunnel	158	
warezmaster	20	1602	20	20	named	17	
warezclient	1020	0	1020	1020	sendmail	17	
spy	2	0	2	2	snmpgetattack	7741	
buffer_overflow	30	30	30	30	ps	16	U2R
loadmodule	9	9	9	9	sqlattack	2	
perl	3	3	3	3			
rootkit	10	10	10	2931			

4.3 Caida DDoS Dataset

CAIDA aims at collecting and sharing of the data for research analysis such as security-related purpose, internet traffic analysis, performance, routing *etc.* It collects data at topologically and geographically different locations and makes the data accessible to the research community. Many categories of datasets are available for various scientific analysis. Internet traces datasets from 2008 to 2016 are available. A most important threat in internet service is Distributed Denial of Service (DDoS) attack [3]. Denial of service attacks try to compromise the availability of the system by keeping it too busy to respond to the service requests of the legitimate users. This type of attack attempts to block access to the targeted server by consuming computing resources on the server and by consuming all of the bandwidth of the network connecting the server to the Internet. This dataset contains approximately one hour of anonymized traffic traces from a DDoS attack on August 4, 2007 (20:50:08 UTC to 21:56:16 UTC). The one-hour trace is split up in 5-minute pcap files. The total size of the dataset is 5.3 GB (compressed; 21 GB uncompressed). Only attack traffic to the victim and responses to the attack from the victim are included in the traces. Non-attack traffic has as much as possible been removed. Traces in this dataset are anonymized using CryptoPAn prefix-preserving anonymization using a single key. The payload has been removed from all packets.These traces can be read with any software that reads the pcap (tcpdump) format, including the CoralReef Software Suite, tcpdump, Wireshark and many others.

4.4 ADFA Linux

Creech and Lu proposed an ADFA Linux (ADFA-LD) cyber security benchmark dataset for evaluation of IDS in the year of 2013 [7]. Ubuntu Linux version 11.04 was used as host Operating System for generating ADFA-LD. It has a higher similarity between normal and attack dataset and contains updated and modern attacks. ADFA-LD is being used by soft computing, cyber security, data mining, machine learning research communities to evaluate the performance of IDS. This dataset provides a contemporary Linux and Windows dataset for evaluation of for host-based intrusion detection system (HIDS).

4.5 UNM Dataset

UNM benchmark dataset was proposed in the year of 2004. System calls data was captured to form this dataset. The dataset contains a very rich set of intrusions such as buffer overflows, symbolic link attacks and trojan programs. Since its scope was very much limited, it cannot replace KDD dataset.

4.6 UNSW-NB15 Dataset

UNSW-NB 15 data set was created in the Cyber Range Lab of the Australian Centre for Cyber Security (ACCS) by the IXIA Perfect Storm tool. This dataset contains a hybrid of normal activities and attack behaviors. Tcpdump tool was used to capture 100 GB of the raw traffic. Twelve algorithms and tools such as Argus, Bro-IDS were used to generate UNSW-NB15. It contains 49 features including a class label [18]. The features, their type and a detailed description is given in Table 3.

A total of 49 attributes determining the features of connections are present for each data instance. The attributes are mixed in nature with some being nominal, some being numeric and some taking on time-stamp values as given in Table 4.

The attributes of the dataset are categorized into 6 broad groups, the details of which are given in Table 5.

This dataset contains a total of 25,40,044 labeled instances, each being labeled either normal or attack. The distribution of connections across the two groups is presented in Table 6.

In total there are nine types of attacks in the dataset in addition to one group representing the normal data. The attacks are categorized as Fuzzers, Analysis, Backdoors, DoS, Exploits, Generic, Reconnaissance, Shellcode and Worms. The details of attacks, subcategory of attacks and the protocols they use are given in Table 7.

5 A Comparison of Data Sets

k-NN classifier with various neighborhood sizes ranging from five to ten was used as a classification model across all the datasets. From the variations of KDD we selected 10 % KDD, Corrected and NSL KDD only. For other datasets since they are having some nominal features like payload, an arbitrary coding was done to convert the nominal values to numeric values. Other datasets having separate files for different attacks and transactions, we merged them into single file to represent each different dataset. Euclidean distance was used to measure the similarity between the instances so as to select the neighbors of a data point. The reasons for using this distance measure is its simplicity and sound mathematical background. As for performance metrics are considered this

Table 3: Features of UNSW-NB15 dataset

SNo.	Name	Type	Description
36	is_sm_ips_ports	Binary	"If source (1) and destination (3)IP addresses equal and port numbers (2)(4) equal thenthis variable takes value 1 else 0"
39	is_ftp_login		If the ftp session is accessed by user and password then 1 else 0.
49	Label		0 for normal and 1 for attack records
7	dur	Float	Record total duration
15	Sload		Source bits per second
16	Dload		Destination bits per second
27	Sjit		Source jitter (mSec)
28	Djit		Destination jitter (mSec)
31	Sintpkt		Source interpacket arrival time (mSec)
32	Dintpkt		Destination interpacket arrival time (mSec)
33	tcprtt		"TCP connection setup round-trip time, the sum of synack and ackdat."
34	synack		"TCP connection setup time the time between the SYN and the SYN_ACK packets."
35	ackdat		"TCP connection setup time the time between the SYN_ACK and the ACK packets."
2	sport	Integer	Source port number
4	dsport		Destination port number
8	sbytes		Source to destination transaction bytes
9	dbytes		Destination to source transaction bytes
10	sttl		Source to destination time to live value
11	dttl		Destination to source time to live value
12	sloss		Source packets retransmitted or dropped
13	dloss		Destination packets retransmitted or dropped
17	Spkts		Source to destination packet count
18	Dpkts		Destination to source packet count
19	swin		Source TCP window advertisement value
20	dwin		Destination TCP window advertisement value
21	stcpb		Source TCP base sequence number
22	dtcpb		Destination TCP base sequence number
23	smeansz		Mean of the ?ow packet size transmitted by the src
24	dmeansz		Mean of the ?ow packet size transmitted by the dst
25	trans_depth		Represents the pipelined depth into the connection of http request/response transaction
26	res_bdy_len		Actual uncompressed content size of the data transferred from the servers http service.
37	ct_state_ttl		No. for each state (6) according to specific range of values for source/destination time to live (10) (11).
38	ct_flw_http_mthd		No. of flows that has methods such as Get and Post in http service.
40	ct_ftp_cmd		No of flows that has a command in ftp session.
41	ct_srv_src		No. of connections that contain the same service (14) and source address (1) in 100 connections according to the last time (26).
42	ct_srv_dst		No. of connections that contain the same service (14) and destination address (3) in 100 connections according to the last time (26).
43	ct_dst_ltm		No. of connections of the same destination address (3) in 100 connections according to the last time (26).
44	ct_src_ ltm		No. of connections of the same source address (1) in 100 connections according to the last time (26).
45	ct_src_dport_ltm		No of connections of the same source address (1) and the destination port (4) in 100 connections according to the last time (26).
46	ct_dst_sport_ltm		No of connections of the same destination address (3) and the source port (2) in 100 connections according to the last time (26).
47	ct_dst_src_ltm		No of connections of the same source (1) and the destination (3) address in in 100 connections according to the last time (26).
1	srcip	Nominal	Source IP address
3	dstip		Destination IP address
5	proto		Transaction protocol
6	state		Indicates to the state and its dependent protocol
14	service		"http ftp smtp ssh dns ftp-data
48	attack_cat		"The name of each attack category. In this data set nine categories *eg* Fuzzers Analysis Backdoors DOS exploits Generic Reconnaissance Shellcode and Worms"
29	Stime	Timestamp	record start time
30	Ltime		record last time

work takes into consideration various measures like Accuracy, Precision and Recall for k-NN for all the data sets.

Table 4: Features type of UNSW-NB15 dataset

SNo.	Feature Type	Count
1	Nominal	6
2	Integer	28
3	Binary	3
4	Float	10
5	Timestamp	2

Table 5: UNSW-NB15 dataset feature categorization

SNo.	Name of the category	Description
1	Flow features	It contains the identifier attributes between hosts such as client-to-serve or server to-client
2	Basic features	It includes the attributes that characterize the connections of protocols.
3	Content features	It contains the attributes of TCP/IP and also contain some attributes of http services
4	Time features	It includes the attributes of time such as round trip time of TCP protocol start/end packet time arrival time between packets *etc.*
5	Additional generated features General purpose features(from number 36 - 40)	Own purpose features which to care for the protocols service.
	Connection features (from number 41- 47)	Built based on the chronological order of the last time feature
6	Labelled Features	It represents the label of the record.

Table 6: Details of events in UNSW-NB15 dataset

Name	Count
Total Number of events	2540044
Normal	2218761
Attacks	321283

Table 7: Categorizations of attacks in UNSW-NB15 dataset

Attack Category	Attack Subcategory	Number of Events
Fuzzers	FTP,HTTP,RIP,SMB,Syslog,PPTP,FTP,DCERPC, OSPF,TFTP,DCERPC,OSPF,BGP	24246
Reconnaissance	Telnet, SNMP, SunRPC Portmapper (TCP) UDP Service , SunRPC Portmapper (TCP) UDP Service, SunRPC Portmapper (TCP) TCP Service, SunRPC Portmapper (UDP) UDP Service, NetBIOS, DNS, HTTP, SunRPC Portmapper (UDP), ICMP, SCTP, MSSQL, SMTP,NETBIOS,DNS	13987
Shellcode	FreeBSD, HP-UX, NetBSD, AIX, SCO Unix, Linux, Decoders, IRIX, OpenBSD, Mac OS X, BSD, Windows, BSDi, Multiple OS, Solaris	1511
Analysis	HTML,Portscanner,Spam	2677
Backdoors	-	2329
DoS	Ethernet, Microsoft Office, VNC, IRC, RDP, TCP, VNC, FTP, LDAP, Oracle, TCP, TFTP, DCERPC, XINETD, IRC, SNMP, ISAKMP, NTP, Telnet, CUPS, Hypervisor, ICMP, SunRPC, IMAP, Asterisk, Browser	16353
Exploits	Evasions, SCCP, SSL, VNC, Backup Appliance, Browser, Clientside Microsoft Office, Interbase, Miscellaneous Batch, SOCKS, TCP, Apache,IMAP, Microsoft IIS, SOCKS, Clientside, Clientside Microsoft Paint, IDS, SSH, ICMP, IDS, DCERPC, FTP, RADIUS, SSL, WINS, Clientside, Clientside Microsoft, POP3, Unix r Service, Cisco IOS, Clientside Microsoft Media Player, Dameware,LPD,MSSQL ,Office Document, RTSP,SCADA,VNC, Webserver,All,LDAP,NNTP,IGMP, Oracle,RDesktop,Telnet,Apache,PHP,SMB,SunRPC,Web Application,DNS,Evasions, RADIUS,BrowserFTP,PPTP,SCCP,SIP,TFTP	44525
Generic	All,SIP,HTTP,SMTP,IXIA,TFTP,SuperFlow,HTTP,TFTP	215481
Worms	-	174

Table 8 given below presents the results of k-NN on all datasets across varying size of neighborhoods. For each run of k-NN, we check the Accuracy, Precision and Recall on all the datasets. As can be seen from the table k-NN has better detection rate and for NSL-KDD data followed by Caida and UNSW-NB respectively in terms of accuracy, precision and recall. The reason for better results on NSL-KDD as compared to other is that this dataset doesn't contain any redundant connections and more over the data is distributed evenly across different classes of attacks and also for normal connections.

In Figure 2 given below a plot of accuracy of k-NN on all the datasets over five different neighborhood sizes is presented. The accuracy is calculated as $Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$. As can be seen from the figure on the average k-NN has better accuracy on NSL-KDD dataset.

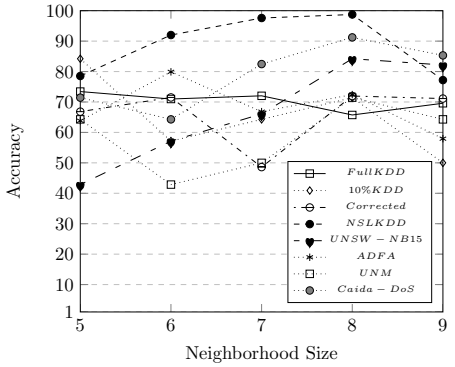


Figure 2: Accuracy

In Figure 2 given below a plot of precision of k-NN on all the datasets over five different neighborhood sizes is presented. The accuracy is calculated as $Precision = \frac{TP}{TP+FP}$. As can be seen from the figure on the average k-NN has better precision on NSL-KDD dataset.

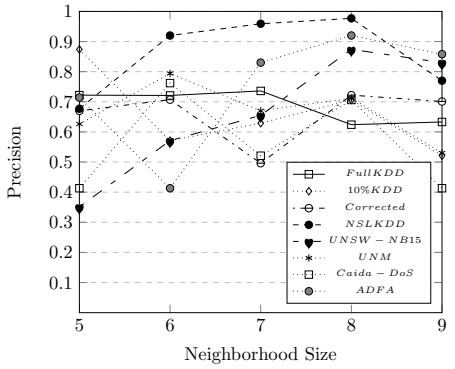


Figure 3: Precision

In Figure 2 given below a plot of Recall of k-NN on all the datasets over five different neighborhood sizes is presented. The accuracy is calculated as $Recall = \frac{TP}{TP+TN}$. As can be seen from the figure like accuracy and precision k-NN has better recall as well on NSL-KDD dataset.

Table 8: k-NN results on datasets

Dataset	Neighborhood														
	5			6			7			8			9		
	Accuracy	Precision	Recall	Accuracy	Precision	Recall	Accuracy	Precision	Recall	Accuracy	Precision	Recall	Accuracy	Precision	Recall
Full KDD99	73.426	0.722	0.734	70.979	0.721	0.710	72.028	0.736	0.720	65.734	0.624	0.657	69.5804	0.633	0.696
Corrected KDD	66.822	0.670	0.668	71.4953	0.707	0.715	48.598	0.496	0.486	71.962	0.722	0.720	71.028	0.701	0.710
NSLKDD	78.537	0.677	0.785	92.000	0.920	0.920	97.592	0.959	0.976	98.750	0.977	0.988	77.193	0.770	0.772
10% KDD	84.210	0.874	0.842	57.142	0.571	0.571	64.285	0.629	0.643	71.428	0.706	0.714	50.000	0.521	0.500
UNSW	42.857	0.351	0.429	57.142	0.571	0.571	66.083	0.655	0.661	84.210	0.874	0.842	82.4561	0.830	0.825
Caida	64.285	0.413	0.643	42.857	0.762	0.351	50	0.521	0.500	71.428	0.706	0.714	64.285	0.413	0.643
ADFA Windows	71.428	0.714	0.714	64.285	0.413	0.643	82.456	0.830	0.825	91.228	0.920	0.912	85.308	0.858	0.853
UNM Dataset	63.827	0.626	0.638	79.906	0.794	0.799	66.822	0.670	0.668	72.429	0.712	0.724	57.943	0.530	0.579

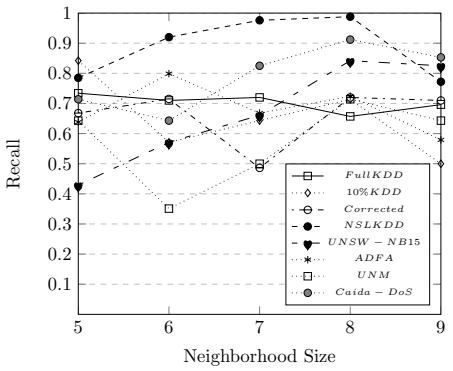


Figure 4: Recall

6 Conclusion

In this work, a thorough review of various benchmark datasets for network intrusion detection is given. The main objective of this work is to provide researchers an idea about what all the datasets are available for the network intrusion detection and what each dataset is comprised of of in terms of features, attacks *etc.* For each dataset, we provided a detailed discussion of its instances, attributes, classes and also the nature of attributes. In addition to that just to get a feel of classification we implemented k-NN classifier employing Euclidean distances over different neighborhoods across all the datasets. The results showed that k-NN performs better and has better Accuracy, Precision and Recall on NSL-KDD, because of the even distribution of instances across various classes and minimal redundancy among the records.

References

[1] R. A. R. Ashfaq, X. Z. Wang, J. Z. Huang, H. Abbas and Y. L. He, "Fuzziness based semi-supervised learning approach for intrusion detection system," *Information Sciences*, vol. 378, pp. 484–497, 2017.

[2] M. C. Belavagi and B. Muniyal, "Performance evaluation of supervised machine learning algorithms for intrusion detection," *Procedia Computer Science*, vol. 89, pp. 117–123, 2016.

[3] M. H. Bhuyan, D. K. Bhattacharyya and J. K. Kalita, "An empirical evaluation of information metrics for low-rate and high-rate DDoS attack detection," *Pattern Recognition Letters*, vol. 51, pp. 1–7, 2015.

[4] K. J. Chabathula, C. D. Jaidhar and M. A. A. Kumara, "Comparative study of principal component analysis based intrusion detection approach using machine learning algorithms," in *3rd International Conference on Signal Processing, Communication and Networking (ICSCN'15)*, pp. 1–6, Mar. 2015.

[5] N. Chand, P. Mishra, C. R. Krishna, E. S. Pilli and M. C. Govil, "A comparative analysis of svm and its stacking with other classification algorithm for intrusion detection," in *2016 International Conference on Advances in Computing, Communication, Automation (ICACCA'16)*, pp. 1–6, Apr. 2016.

[6] M. H. Chen, P. C. Chang and J. L. Wu, "A population-based incremental learning approach with artificial immune system for network intrusion detection," *Engineering Applications of Artificial Intelligence*, vol. 51, pp. 171–181, 2016.

[7] G. Creech and J. Hu, "Generation of a new ids test dataset: Time to retire the kdd collection," in *IEEE Wireless Communications and Networking Conference (WCNC'13)*, pp. 4487–4492, 2013.

[8] A. S. Eesa, Z. Orman and A. M. A. Brifcani, "A novel feature-selection approach based on the cuttlefish optimization algorithm for intrusion detection systems," *Expert Systems with Applications*, vol. 42, no. 5, pp. 2670–2679, 2015.

[9] K. Gai, M. Qiu, L. Tao and Y. Zhu, "Intrusion detection techniques for mobile cloud computing in heterogeneous 5G," *Security and Communication Networks*, vol. 9, no. 16, pp. 3049–3058, 2016.

[10] Y. Hamid, M. Sugumaran and L. Journaux, "A fusion of feature extraction and feature selection technique for network intrusion detection," *International Journal of Security and Its Applications*, vol. 10, no. 8, pp. 151–158, 2016.

[11] S. Janakiraman and V. Vasudevan, "ACO based distributed intrusion detection system," *International Journal of Digital Content Technology and its Applications*, vol. 3, no. 1, pp. 66–72, 2009.

[12] G. Kim, S. Lee and S. Kim, "A novel hybrid intrusion detection method integrating anomaly detection with misuse detection," *Expert Systems with Applications*, vol. 41, no. 4, pp. 1690–1700, 2014.

[13] L. Koc, T. A. Mazzuchi and S. Sarkani, "A network intrusion detection system based on a hidden naive bayes multiclass classifier," *Expert Systems with Applications*, vol. 39, no. 18, pp. 13492–13500, 2012.
[14] H. J. Liao, C. H. R. Lin, Y. C. Lin and K. Y. Tung, "Intrusion detection system: A comprehensive review," *Journal of Network and Computer Applications*, vol. 36, no. 1, pp. 16–24, 2013.
[15] W. C. Lin, S. W. Ke and C. F. Tsai, "Cann: An intrusion detection system based on combining cluster centers and nearest neighbors," *Knowledge-Based Systems*, vol. 78, pp. 13–21, 2015.
[16] G. Liu, Z. Yi and S. Yang, "A hierarchical intrusion detection model based on the PCA neural networks," *Neurocomputing*, vol. 70, no. 7, pp. 1561–1568, 2007.
[17] K. Malialis and D. Kudenko, "Distributed response to network intrusions using multiagent reinforcement learning," *Engineering Applications of Artificial Intelligence*, vol. 41, pp. 270–284, 2015.
[18] N. Moustafa and J. Slay, "The evaluation of network anomaly detection systems: Statistical analysis of the UNSW-NB15 data set and the comparison with the KDD99 data set," *Information Security Journal: A Global Perspective*, pp. 18-31, 2016.
[19] S. Mukkamala, A. H. Sung and A. Abraham, "Intrusion detection using an ensemble of intelligent paradigms," *Journal of network and computer applications*, vol. 28, no. 2, pp. 167–182, 2005.
[20] A. A. Orunsolu, A. S. Sodiya, A. T. Akinwale, B. I. Olajuwon, M. A. Alaran, O. O. Bamgboye, and O. A. Afolabi, "An empirical evaluation of security tips in phishing prevention: A case study of nigerian banks," *International Journal of Electronics and Information Engineering*, vol. 6, no. 1, pp. 25-39, 2017.
[21] A. A. Orunsolu, A. S. Sodiya, A. T. Akinwale, B. I. Olajuwon, "An anti-phishing kit scheme for secure web transactions," *International Journal of Electronics and Information Engineering*, vol. 6, no. 2, pp. 72–86, 2017.
[22] S. Peddabachigari, A. Abraham, C. Grosan and J. Thomas, "Modeling intrusion detection system using hybrid intelligent systems," *Journal of Network and Computer Applications*, vol. 30, no. 1, pp. 114–132, 2007.
[23] R. R. Reddy, "Network intrusion anomaly detection using radial basis function networks," *International Journal of Advanced Research in Computer Science*, vol. 8, no. 3, pp. 1011–1014, 2017.
[24] S. Shamshirband, A. Patel, N. B. Anuar, M. L. M. Kiah and A. Abraham, "Cooperative game theoretic approach using fuzzy q-learning for detecting and preventing intrusions in wireless sensor networks," *Engineering Applications of Artificial Intelligence*, vol. 32, pp. 228–241, 2014.
[25] A. Tayal, N. Mishra and S. Sharma, "Active monitoring & postmortem forensic analysis of network threats: A survey," *International Journal of Electronics and Information Engineering*, vol. 6, no. 1, pp. 49–59, 2017.

Biography

Yasir Hamid received his Master's degree in Computer Applications from University of Kashmir in the year 2014. He is currently as a Ph.D Scholar in Department of Computer Science and Engineering, Pondicherry Engineering College, Puducherry. His areas of interests are Machine Learning, Network Security, Non Linear Dimension Reduction and data visualization.

V. R. Balasaraswathi recieved her M.E Degree in Computer Science from Anna University Chennai, currently she is working towards Ph.D degree at Dept. Computer Science and Engineering, Pondicherry Engineering College.

Ludovic Journaux received his PhD in image processing and computer sciences from the University of Burgundy (France) in 2006. He is currently working as associate professor at Agrosup Dijon and is a member of LE2I laboratory (UMR 6306) : Laboratory of Electronics, Computer Sciences and Images. His research interests include image processing, data mining, statistical analysis, artificial intelligence and classification.

M. Sugumaran received his M.Sc degree in mathematics from University of Madras in 1986, M.Tech degree in computer science and data processing from Indian Institute of Technology, Kharagpur, India in 1991 and obtained his Ph.D from Anna University, Chennai in 2008. He is currently working as Professor and Head of Computer Science and Engineering at Pondicherry Engineering College, India. His areas of interests are theoretical computer science, analysis of algorithms, parallel and distributed computing and spatial-temporal data.

An Efficient Practice of Privacy Implementation: Kerberos and Markov Chain to Secure File Transfer Sessions

Fadi Al-Ayed, Chunqiang Hu and Hang Liu
(Corresponding author: Fadi Al-Ayed)

School of Electrical Engineering and Computer Science, The Catholic University of America
Washington, DC 20064, USA
(Email: 93alayed@cua.edu)

(Received May 1, 2017; revised and accepted July 25, 2017)

Abstract

Kerberos is a widely used authentication and authorization protocol that allows a client to communicate with servers to authenticate mutually, and obtain authorization and tickets to access application services securely with encryption over a non-secure network. It is designed to provide strong confidentiality, integrity, and authentication through symmetric key cryptography. However, Kerberos system itself needs to be monitored and protected. In this paper, we present the implementation of an intrusion detection system (IDS) that monitors a Kerberos network, and reports and responds to malicious activities or policy violations. The designed IDS system performs anomaly activity detection using a Markov fingerprinting scheme that builds a Markov chain model of the normal Kerberos session messages and applies the machine learning technique to detect deviations from the model of normal traffic. We mainly focus on detecting and averting intrusion attempts on File Transfer Protocol (FTP) applications for corporate systems. The proposed scheme can be extended to support other applications on a Kerberos network. The results of our experiments show that implementing Markov fingerprinting with Kerberos can improve the security in terms of prevention and detection of malicious behaviors.

Keywords: Encrypted Traffic; File Transfer Protocol; Kerberos; Markov Model

1 Introduction

Secure data transfer sessions in corporate networks using File Transfer Protocol (FTP) is becoming a vital process. The traditional authentication scheme that simply verifies a user through a login name and password transmitted across the network with plain text is vulnerable in terms of security and privacy because information sent across the network may be intercepted and subsequently used to impersonate the user maliciously. In addition, different services are provided to multiple users on the corporate networks, which requires the ability to identify the authorization of the user to access different services [8,16,17]. Kerberos is a network authentication and authorization protocol that allows clients and servers to authenticate each other, and a client to obtain authorization and tickets to access application services securely with encryption over a non-secure network. It is designed to provide strong confidentiality, integrity, and authentication through symmetric key cryptography [11, 15, 30]. The main idea under Kerberos is to send a hash of the user's password over the network instead of the actual password itself. The password is checked on both sides of the connection for authentication and authorization, and is also used as a key for symmetric hash encryption. A timestamp is used with the assumption that clients have loosely synchronized time. Kerberos enables a client to be authenticated and authorized to access multiple servers spontaneously. It also has very strong protection against eavesdropping and replay attacks.

Kerberos has been widely used. For example, Windows networking uses Kerberos as its preferred authentication method, with which a client joins a Windows domain, proves its identity for authentication, and obtains the authorized services in the domain and all domains with trust relationships to that domain.

Note that Secure Sockets Layer (SSL), or Transport Layer Security (TLS) is another known network security protocol that provides privacy and data integrity between two communicating computer applications [3]. SSL is a standard security technology for authentication and establishing a secure link between the web server and browser. However, we consider enterprise applications in this paper for which Kerberos provides a simple and general framework for securing the network applications.

Kerberos system itself needs to be monitored and pro-

tected. In this paper, we present an implementation of an intrusion detection system (IDS) that monitors a Kerberos network, and reports and responds to malicious activities or policy violations. The designed IDS system detects anomaly activities using a Markov chain to build a stochastic model to represent Kerberos session states. Markov chains [2,9] have been utilized in information processing by taking an advantage of its ability to capture statistical regularities in the behavior of systems and allowing state estimation and pattern recognition. The session states reflects the Kerberos protocol and messages in single-directional traffic flows from client to server or from server to client for an application. Based on the Markov chain model of a normal Kerberos session for a particular application, we perform anomaly detection by detecting deviations from the model of normal traffic using a learning technique, which forms Markov fingerprinting. We seek to implement the state estimation and pattern recognition provided by the Markov chain model to come up with a more pro-active approach for the detection and avoidance of malicious behaviors in Kerberos protocol communications. By establishing a fingerprint through identifying the possible sequence of messages, we would be able to detect abnormal behaviors that might be attempts of intrusion. We mainly focus on detecting and averting intrusion attempts on the FTP type of applications as it is popular for corporate systems. The proposed scheme can be extended to support other applications on a Kerberos network.

The remainder of this paper is organized as follows: Section 2 discusses the related work. Section 3, presents the design methodology. Experimental results are given in Section 4, and Section 5 concludes the paper.

2 Related Work

Computing and software development have evolved from the noble objectives of providing solutions for daily problems to more malicious intents. Disrupting and stealing, or harming, have been obfuscations items for quite a while and calls for stricter implementations on data security. Computer systems are still human driven, and as such, there is an element of unpredictability. This then led to quite a growing effort to infuse some pattern and behavioral recognition capabilities in the applications and the protocols they are running on. In the study of intrusion detection, and probabilistic techniques have been used which would be represented as decision trees.

Ye *et al.* made several tests for Markov chain which was applied to set of computer audit data to investigate the frequency and ordering property of the information [28]. The study gave answers to questions on which properties were critical to detect intrusions techniques that were based on the frequency property provided solution that resulted to good intrusion detection.

Abraham *et al.* conducted a study of implementing an attack graph that was very similar in characteristics to a decision tree [2]. With the combination of vulnerabilities observed in a network configuration, several scenarios were built where an attacker can reach the goal state [10]. Based on the attack graph, Markov chain simulation was conducted.

Other studies focused on the encrypted network traffic flow and their classifications. Korczynski *et al.*, on traffic encryption, postulated a method that uses the size of first few packets of data as a basis to enable early application recognition [2, 12, 19, 23]. Another method tried identification of SSL/TLS encrypted application layer protocols using a signature-based and a flow-based statistical analysis process.

Signature recognition technique is the term that Ye *et al.* used for the ability to identify anomalies in behavior and signal intrusions. He developed an anomaly detection process that corresponds to the norm profile of a temporal behavior via Markov model [5, 27]. Qassim *et al.* [21] proposed a network anomalies classifier which utilizes machine learning to classify activities detected and to monitor network behavior by a packet header based anomaly detection system.

The aforementioned works have all been centered on the utilization of the Markov chain to be able to come up with signatures of behavior that can be used to identify intent. Our work, however, is an advance approach in this direction to enhance the traditional way which would be revolving around the same concept.

3 Methodology

In this section, we describe the implementation of the Markov chain based intrusion detection scheme for Kerberos sessions.

3.1 Kerberos Overview

As shown in Figure 1, a Kerberos system consists of Authentication Server (AS), Ticket Granting-Ticket (TGT), and application server [24]. A client sends a Ticket-Granting-Ticket (TGT) request to the AS, the AS verifies the access right in database based on the user's ID, generates a TGT and session key, and send them to the client in a reply message. The TGT is a token which enables the user to request access to application services without re-supplying its credentials. The TGT is encrypted by a TGS key that TGS knows, but the user would not know. The session key is encrypted using a key derived from the user's password. The client will ask for the user's password and use it to decrypt the incoming reply message to get the client-TGS session key after it receives the reply from the AS. When the client first attempts to access an application service, it sends the request message for the application service granting ticket to the TGS. This request message contains TGT and an authenticator that contains user ID, network address, and timestamp, and is encrypted by the client-TGS session key.

The TGS decrypts the TGT and authenticator, verifies the request, generates a granting ticket for the particular requested application server if it is authorized, and a client-application server session key, and sends the reply to the client. The ticket is encrypted by an application server key that knows by the application server but not the client, and the client-application server session key is encrypted with the client-TGS session key. After the client receives the reply message from TGS, it obtains the application service granting ticket. To access the application service, the client sends the request message to the application server, which contains the application service granting ticket and authenticator. The authenticator is encrypted with the client-application server session key. The application server verifies the ticket and authenticator, and grants access to the application service according to the authorization data specified in the ticket. The connection between the user and the application service is thus established. If mutual authentication is required, the server returns an authenticator to the client. Table 1 summarizes the Kerberos messages exchanges, and Table 2 presents the notation of the negotiation parameters between the server and client [24, 26].

Table 1: Summary of Kerberos message exchanges

(1) C → AS	$Options \parallel ID_c \parallel Realm_c \parallel ID_{tgs} \parallel Times \parallel Nonce_1$
(2) AS → C	$Realm_c \parallel ID_C \parallel Ticket_{tgs} \parallel \mathrm{E}(K_c, \parallel K_{c,tgs} \parallel Times \parallel Nonce_1 \parallel Realm_{tgs} \parallel ID_{tgs}])$
	$Ticket_{tgs} = \mathrm{E}(K_{tgs}, [Flags \parallel K_{c,tgs} \parallel Realm_c \parallel ID_C \parallel AD_C \parallel Times])$

(a)Authentication Service exchange to obtain ticket-granting ticket

(3) C → TGS	$Options \parallel ID_v \parallel Times \parallel Nonce_2 \parallel Ticket_{tgs} \parallel Authenticator_c$
(4) TGS → C	$Realm_c \parallel ID_C \parallel Ticket_v \parallel \mathrm{E}(K_{c,tgs}, [K_{c,v} \parallel Times \parallel Nonce_2 \parallel Realm_v \parallel ID_v])$
	$Ticket_{tgs} = \mathrm{E}(K_{tgs}, [Flags \parallel K_{c,tgs} \parallel Realm_c \parallel ID_C \parallel AD_C \parallel Times])$
	$Ticket_v = \mathrm{E}(K_v, [Flags \parallel K_{c,v} \parallel Realm_c \parallel ID_C \parallel AD_C \parallel Times])$
	$Authenticator_c = \mathrm{E}(K_{c,tgs}, [ID_C \parallel Realm_c \parallel TS_1])$

(b)Ticket-Granting Service exchange to obtain service-granting ticket

(5) C → V	$Options \parallel Ticket_v \parallel Authenticator_c$
(6) V → C	$\mathrm{E}_{K,c,v}[TS_2 \parallel Subkey \parallel Seq\#]$
	$\mathrm{Ticket}_v = \mathrm{E}(K_v, [\mathrm{Flags} \parallel K_{c,v} \parallel Realm_c \parallel ID_C \parallel AD_C \parallel Time_s])$
	$Authenticator_c = \mathrm{E}(K_{c,v}, [ID_C \parallel Realm_c \parallel \mathrm{TS}_2 \parallel Subkey \parallel Seq\#])$

(c) Client/Server Authentication exchange to obtain service

Table 2: Notation

Notation	Description
AS	Authentication Server
C	Client
V	Server
ID_C	Identifier of user on C
ID_V	Identifier of V
P_C	Password of user on C
AD_C	Network address of C
$\parallel$	String concatenation operation
K_V	Secret encryption key shared by AS and V

The remaining attributes are described as follows [24], *Realm*: Indicates realm of user. *Options*: Used to request that specific flags be sent in the returned ticket. *Times*: Used for time settings in the ticket e.g. desired start time, and expiration time for the requested ticket. *Nonce*: A random value to be repeated in message to assure that the response is fresh and have not been replayed by an opponent. *Subkey*: This field is omitted, since the session key from the ticket (KC,V) is used. *Seq*: This optional field that specifies the starting sequence number to be used by the server for messages sent to the client during this session.

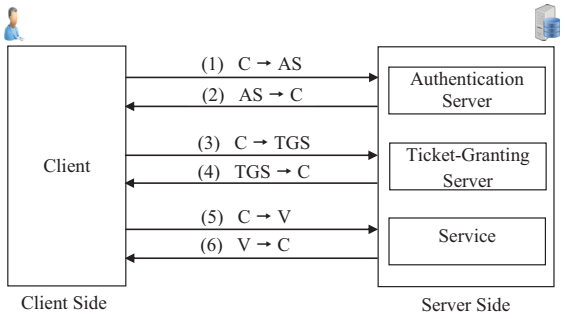


Figure 1: Overview of kerberos

3.2 Markov Model Fingerprint

In this section, we illustrate an approach based on Markov chains to model possible sequences of message types observed in a single-directional Kerberos session.

Consider a discrete-time random variable X_t for any $t = t_0,\ t_1, \ldots, t_n \in T$ such that X_t where $1 \leq t \leq n$. It takes values $i_t \in \{1, \ldots, s\}$ that corresponds to the observed Kerberos message types during a session. Assuming that X_t is the first-order Markov chain [12, 13, 29], and P is denoted as the transition matrix then:

$$P\left(X_t = i_t \mid X_{t-1} = i_{t-1}, X_{t-2} = i_{t-2}, \ldots, X_0 = i_0\right) = P\left(X_t = i_t \mid X_{t-1} = i_{t-1}\right). \tag{1}$$

Assuming further that the Markov chain is homogeneous, i.e. state transition is time-invariant:

$$P\left(X_t = i_t \mid X_{t-1} = i_{t-1}\right) = P\left(X_t = j \mid X_{t-1} = i\right) = p_{i,j} \tag{2}$$

Using the transition matrix:

$$P = \begin{bmatrix} P_{1,1} & P_{1,2} & \cdots & P_{1,s} \\ P_{2,1} & P_{2,2} & \cdots & P_{2,s} \\ \cdots & \cdots & \ddots & \vdots \\ P_{s,1} & P_{s,2} & \cdots & P_{s,s} \end{bmatrix}$$

Where:

$$\sum_{j=1}^{s} P_{i,j} = 1.$$

The function:

$$Q = [q_1, q_2, \cdots, q_s], \tag{3}$$

is denoted as the Initial Probability Distribution (IPD) where q_i=P(X_t=i) at time t=0, and the following function:

$$W = [w_1, w_2, \cdots, w_s], \tag{4}$$

is denoted as the Exit Probability Distribution (EPD) where *wi* characterizes the probability that the Kerberos session finishes when if it is in state i at time tn. However, both initial and exit probability distributions are independent of the Markov chain.

Lastly, the probability that a sequence of states $X_1,\ldots,X_T$ corresponding to a single Kerberos session occurs is denoted by:

$$P(X_1, \ldots, X_T) = q_i \prod_{t=2}^{T} P_{i_{t-1}}, i_t \times W_{iT}. \tag{5}$$

The observed transition probability matrices and the IPD for the Kerberos sessions can be used by the Markov Classifier to classify traffics and identify traffic anomaly as described below. For example, an attacker may steal or forge a Kerberos TGT and attempts to gain the application services with is called Golden Ticket attack [8]. Such a Golden Ticket attack can be detected based on the Kerberos traffic anomaly it creates, an attacker sends a valid TGT to the TGS with no prior successful AS requests to obtain a TGT.

3.2.1 Markov Classifier (MC)

The Markov Classifier (MC) uses a first-order homogeneous Markov chain to build a stochastic model that reflects the Kerberos session states. A Kerberos session model is obtained per flow direction that corresponds to each transaction or process used. The decimal codes shown in Table 3 are used to represent the states during a Kerberos session.

Table 3: Kerberos states codes

SID	Description
20:	Username Exist ($Username\_Exits$)
21:	Check Active Director (AD_Check)
22:	Request for TGT ($Ticket\_request$)
23:	Generation of TGT and Session Key ($TGT\_SessionKey$)
24:	Request for Service Ticket ($GS\_Success$)
25:	Generation of Service Ticket and Another Session Key ($e\_h\_Session\_Key$)
26:	Present Service Ticket to Server (Tkt_http)
27:	Authenticate and Approve the use of Service ($Kerberos\_Success$)
28:	Authenticated Users ($Kerberos\_Success = 1$)
29:	Check for Attackers ($Alert\_Exit$)
30:	Unknown Users ($Username\_Exits = 0$)

As shown above, the state of check username exist decimal code is 20: and decimal code 21: is for checking the active directory of the clients (AD_Check). The initial user authentication request ($Ticket\_request$) is represented by a decimal code of 22: The reply of the ($TGT\_SessionKey$) is denoted by 23: which contains the TGT and the client-TGS session key. Decimal code 24: is the request to the TGS ($GS\_Success$) by the client for an application service ticket. The TGS reply ($e\_h\_Session\_Key$) to the request is denoted by 25: which contains the application service ticket and application service session key created by TGS. Decimal code 26: is the client's request to the application server (Tkt_http), which includes the ticket and client authenticator to access the application service. The application server that authenticates the user which is represented by the decimal code 27: The process may also respond ($Kerberos\_Success$) to perform mutual authentication and indicate completion of the process with a code 28: In addition, the alert state decimal code is 29: and the state of unknown users decimal code is 30.

3.2.2 Learning and Ranking Mechanisms

We utilize the rule-based learning algorithm to classify the Kerberos session Markov state parameters for different types of users [4, 18, 31]. Table 4 shows the variables.

Table 4: Notation and description of each variable

Notation	Description
$q_{(n)}$	Represents each Kerberos parameter
q	Sum of indexed Kerberos values
$x_{(m)(n)}$	x=values in session, n=counter, and m=user type
$y^{(n)}$	Sum of previous session
?	Queries of summed sessions
h_x	Results of desired queries

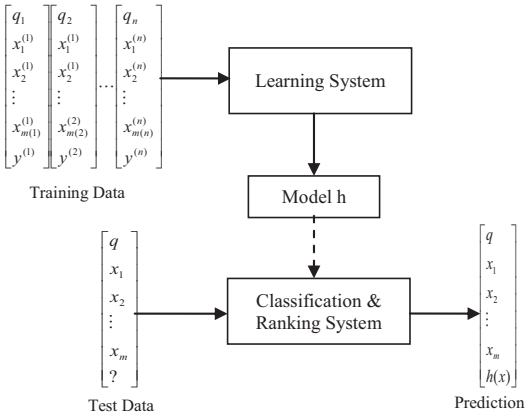


Figure 2: Classification architecture

Figure 2 shows the learning and classification architecture. To build the Markov models of the Kerberos session behaviors for different type of users, the Markov classifier is first trained. We use the Kerberos session data for normal users and known attacks to train the MC and build the Markov models. The messages of Kerberos sessions are collected and pre-processed. Essential features that can differentiate one session from the other such as user IDs, IP addresses and ports are identified. The MC

analyzes the messages, calculates the Markov model parameters for each type of users, and builds the models for normal authorized users and known attacks as discussed later in this section. The trained classifier is used to detect malicious activities on the test data. The data captured in the test phase is classified based on the Markov chain state transition fingerprints of different types of users, normal users and adversaries which will have different parameters of the Markov models. The decision process for session classification and anomaly detection is based on both the likelihood criterion and the likelihood ratio criterion as described in the next subsection.

Figure 3 illustrates an overview of the MC-based classifier operations. It consists of data flow which records every single activities messages exchanged during Kerberos sessions into a log and control flow to estimate the execution of the following operations: likelihood criterion and the likelihood ratio criterion.

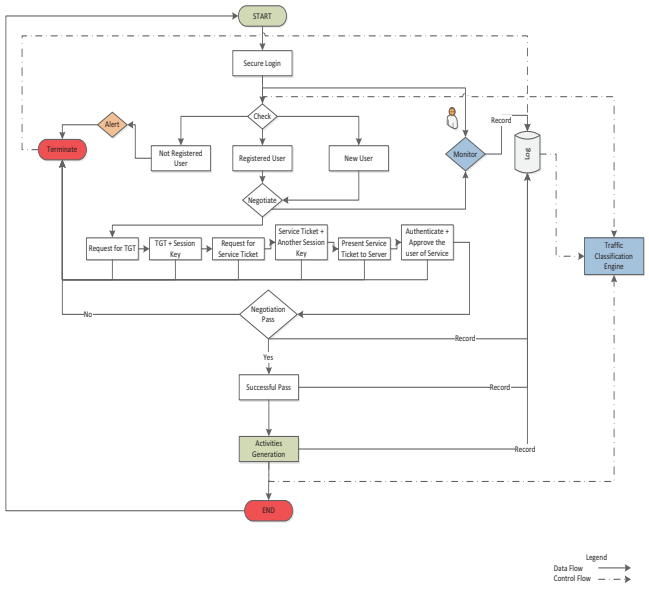


Figure 3: Overview of operational architecture

To begin with data flow (new session), a user is required to enter the granted username and password on the login page. The client has three attempts to login. If the user fails the attempts, the account will automatically be locked out and a notification will be sent to both original client and system administrator. If the client has been successfully logged in, the system checks for the previous user's information recorded in the database for further security purpose. Each client should have one secure login page, in case the client changes the workstation, another secure login page must be generated by the system administrator, otherwise it begins a Kerberos negotiation process. For each user attempt: first, second or third, a new Kerberos session is created. Hence, a new Kerberos ID is related to each attempt. The negotiation process is as flows: the client requests for TGT, if there is no response from either the client or the server, the data flow connection will be terminated and the data of this activity is recorded into a log. The mechanism process applies the same for other Kerberos parameters. More precisely, every single activity in data flow is recorded into a log either if its succeed or failed. The determination of data flow is maintained by control flow for further execution process to classify a session. The execution of control flow will be studied in the next section.

Algorithm 1 shows how the MC classifier computes the transition matrix based on the messages exchanged during Kerberos sessions.

Algorithm 1 Working of Markov

1: Begin
2: Choose Initial parameter estimates

$$Q = [q_1, q_2, \cdots, q_s],$$

3: **if** Username Exit is invalid **then**
4: Drop the message
5: **else**
6: Compute:

$$P\left(X_t = i_t \mid X_{t-1} = i_{t-1}, X_{t-2} = i_{t-2}, \ldots, X_0 = i_0\right) =$$

$$P\left(X_t = i_t \mid X_{t-1} = i_{t-1}\right).$$

7: Compute:

$$P(X_1, \ldots, X_T) = q_i \prod_{t=2}^{T} P_{i_{t-1}, i_t} \times W_{iT}.$$

8: Verify the completed negotiated messages:
9: **if** negotiated messages = True **then**
10: Accept the user session
11: **else**
12: Drop the message
13: Compute:

$$W = [w_1, w_2, \cdots, w_s],$$

14: **end if**
15: **end if**
16: End

3.2.3 Session Classification and Anomaly Detection

In order to classify a session and detect potential attacks, the trained MC classifier will process the network traffic and extract the message exchange for a Kerberos session based on the user ID, IP address and port. It analyzes the messages exchanged for a session and makes a classification decision. The message exchanges behavior and Markov model parameters may be known for some kinds of attacks, but unknown for new types of attacks. Thus, the decision process is based on two test criteria [6, 7, 14, 25]:

1) The likelihood to be a normal authorized session;
2) The likelihood ratio of being a normal session to being abnormal session. If one tests fails, it is assumed that the session is potentially an abnormal session.

For likelihood test, the null and alternative hypotheses are:

H_0**:** A session is normal (Null Hypothesis);

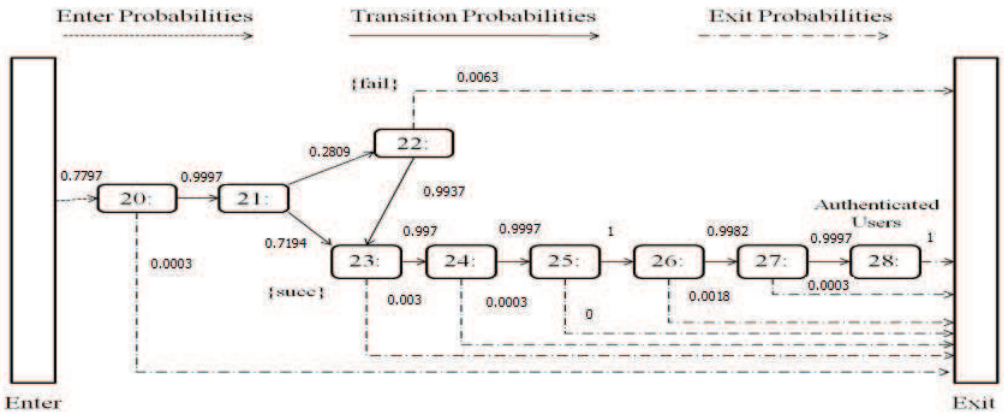


Figure 4: Parameters of the fingerprint for normal users

H_1: A session is abnormal (Alternative Hypothesis).

Given the input data sequence corresponding to the observed Markov states during a session,$\{x_1, \ldots, x_T\}$, the likelihood function of being a normal session is equal to:

$$L(H_0|x_1, x_2, \ldots, x_T) = P(x_1, x_2, \ldots, x_T|H_0) \quad (6)$$

The decision rules are if $L(H_0|x_1, x_2, \ldots, x_T) \geq \Delta$, don't reject H_0; if $L(H_0|x_1, x_2, \ldots, x_T) < \Delta$, reject H_0. We can select an appropriate value of likelihood criterion Δ. The impact of the likelihood criterion Δ will be studied in the next session.

For likelihood ratio test, we consider the null and alternative hypotheses which are:

H_0: A session is normal (Null Hypothesis);

H_1: A session is one of the known attacks (Alternative Hypothesis).

Assume that Markov model parameters of the Kerberos sessions for a set of attack types, $\{\Omega_1, \Omega_2, \ldots, \Omega_k\}$ have been known. Given the observed Markov states during a session, $\{x_1, \ldots, x_T\}$, we can find the maximum value of the likelihood function of being one of the known attack types that is:

$$L(H_1|x_1, x_2, \ldots, x_T) = \arg_{\Omega_i} \max L(\Omega_i|x_1, \ldots, x_T) \quad (7)$$

The likelihood being a type of attacks is the probability of the Markov state sequence computed over such a type of attacks, $L(\Omega_i|x_1, \ldots, x_T) = P(x_i, \ldots, x_T|\Omega_i)$. Then, the likelihood ratio is:

$$\Lambda = \frac{L(H_0|x_1, x_2, \ldots, x_T)}{L(H_1|x_1, x_2, \ldots, x_T)} = \frac{L(H_0|x_1, x_2, \ldots, x_T)}{\arg_{\Omega_i} \max L(\Omega_i|x_1, \ldots, x_T)}. \quad (8)$$

The likelihood ratio test provides the decision rule as follows:

$$\begin{cases} \text{If } \Lambda \geq \Gamma, & \text{do not reject } H_0 \\ \text{If } \Lambda < \Gamma, & \text{reject } H_0 \end{cases}$$

The values Γ is chosen to obtain a specified significance level, and its impact will be studied in the next section.

Finally, we consider H_0 is rejected if either the above likelihood test or the likelihood ratio test fails.

4 Experimental Results

In this section, we show the experimental results and evaluate the performance of the Markov chain model based on anomaly detection scheme.

4.1 Examples of Markov Fingerprints

We have conducted the experiments to obtain Markov chain models for two different types: normal users and attackers. We have also examined the scheme to detect a special type of attack which is known as Golden Ticket attack.

Figure 4 demonstrates the observed parameters of the Markov chain model for normal users. The change of state from decimal code 20: which is check *Username_Exits* to decimal code 21: which is *AD_Check* is probable by 99.97% of sessions, whereas 0.03% are sessions representing failure and closing prior to authentication process. A TGT is set to be valid for 10 hours in our experiments. Hence, this would prevent the client from having any further login issues, instead of going through the entire process of requesting for a new ticket. It is managed through the Active Directory which keeps active record of the clients who logged in within the 10 hours. Whenever a user logs in, the Active Directory (AD) is checked for record, if a TGT exists, the user is allowed to proceed to the server; otherwise, the user has to request for a TGT. From the above Figure 4, 28.1% out of 99.97% requires a new request for TGT and 71.9% out of 99.97% does not. However, at state decimal code 22: which is *Ticket_request* has 99.37% probability to continue to the next state of decimal code 23: which is *TGT_SessionKey*. Furthermore, the flow from the state of decimal code 23: to the state of decimal code 24: which is *GS_Success* is probable by 99.7% of sessions. Consequently, successful authentication state which is decimal code 27: has an 99.97% probability to continue to the next state of decimal code 28: which includes all the successful Kerberos sessions. We observe that the success rate at each state for the normal users is considerably high.

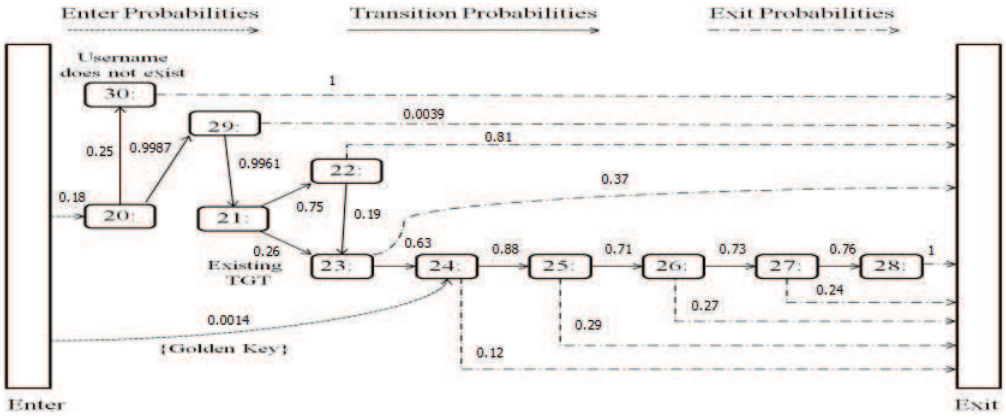


Figure 5: Parameters of the fingerprint for attackers

Figure 5 demonstrates the observed parameters of the Markov chain model for abnormal users (attackers).

As shown above in Figure 5, the change of state from decimal code 20: which is check *Username_Exits* to decimal code 29: which is *Alert_Exit* is probable by 99.87%. At the Alert state, past record of the user's session is checked including the number of attempts compared with the previous successful attempts. Meanwhile, the change of state from decimal code 20: to decimal code 30: which is for unknown users, is probable by 25% which no record exists. We observe that the success rate at each state for the attackers is much lesser.

Golden Ticket Attack, is a special type of attack particularly on the Kerberos approach. In Kerberos, every TGT is valid for certain period of time (10 hours in our case). A golden ticket attacker with get hold of this TGT, has the ability to edit it and reuse it. Typically, the attacker either increases the validity of the ticket or increases its privileges or uses TGTs of deleted users. Since privileges are not used in our case, we have considered the validity of the ticket and TGTs of deleted users. These types of golden attacks are considered on the following criteria:

- Attempts of validity is increased of existing user.
- Attempts of validity is increased of deleted user.

Basically, the attackers edit the TGTs. So there is no original Ticket request associated with the TGTs. To detect the attacks, we examine a Ticket request associated with every TGT, if it is not available, corresponding session which is probable by 00.14% of golden ticket attacks.

This demonstrates the observed parameters of the Markov chain model for golden ticket attacks, with which the adversaries have forged Kerberos TGTs. The adversaries can control every aspect of the forged ticket including the Ticket's user identity, permissions and ticket life time. Since the adversaries use the forget TGT in a TGS-REQ message, they do not need to go through the authentication transaction to create the TGT, or the authentication and authorization transaction does not match the TGT [8].

4.2 Performance of MC-based Classification and Anomaly Detection

We have established a dataset, which was used to evaluate our scheme, consists of 4330 records represents the number of flows identified in Kerberos data.

We conducted the experiments in order to evaluate and to assess the performance of the Markov based classification and anomaly detection: the True Positive Rate denoted as TPR and False Positive Rate denoted as FPR, respectively. True Positive occurs as a class of user Kerberos sessions is correctly classified as the given user session. False Positive arises as another class of user sessions is incorrectly classified as the given class of session. More specifically, the True Positive represents the number of actual attacks classified as attacks, False Positive corresponds to the number of actual normal sessions classified as attacks, True Negative represents the number of actual normal sessions classified as normal sessions, and False Negative corresponds to the number of actual attacks classified as normal sessions [4, 12]. The True Positive Rate (TPR) also represents the detection rate, and can be calculated as [1, 20, 22]:

$$TPR = \frac{TP}{TP + FN} \tag{9}$$

The False Positive Rate (FPR) can be calculated as:

$$FPR = \frac{FP}{FP + TN} \tag{10}$$

4.3 Classification Results

We report the classification results of MC classification experiments below. Table 5 shows the results for MC on the dataset.

Using Equation (9), the TPR result is 0.9801, and Equation (10) for FPR which is 0.0264. We notice that the TPR is very large with relatively small rate of FPR.

Figure 6 shows the performance results of evaluating the trade-offs between the prospective true positive rate and false positive rate. The higher the area under

Table 5: Comparision standard martix

		Predicted Label	
		Positive	Negative
Actual Class	Positive	True Positive (3352)	False Negative (68)
	Negative	False Positive (24)	True Negative (886)

the curve top left corner, the better performance of the scheme.

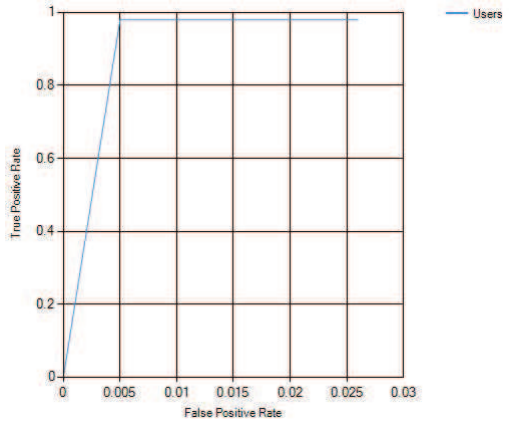


Figure 6: True positive rate vs. False positive rate

A major improvement is noticed with the true positive rate as the curve is very close to the perfect classification point (0,1). Thus, the Markov based classification is very effective.

5 Conclusion

In this paper, we have presented the implementation and evaluation of a Markov chain model based scheme for classification and anomaly detection of Kerberos sessions. The fingerprints of the Kerberos messages are classified based on first-order homogeneous Markov chain and used to detect the anomaly. The evaluation results show that implementing Markov fingerprinting with Kerberos can improve the security in terms of detection of malicious behaviors.

References

[1] A. Abraham, C. Grosan and C. Martin-vide, "Evolutionary design of intrusion detection programs," *International Journal of Network Security*, vol. 4, no. 3, pp. 328–339, 2007.

[2] S. Abraham and S. Nair, "Cyber security analytics: A stochastic model for security quantification using absorbing markov chains," *Journal of Communications*, vol. 9, no. 12, pp. 899–907, 2014.

[3] F. Al-Ayed and H. Liu, "Synopsis of security: Using kerberos method to secure file transfer sessions," in *International Conference on Computational Science and Computational Intelligence*, pp. 1016–1021, 2016.

[4] M. Ambusaidi, X. He, P. Nanda and Z. Tan, "Building an intrusion detection system using a filter-based feature selection algorithm," *IEEE Transactions on Computers*, vol. PP, no. 10, pp. 2986–2998, 2016.

[5] S. Babu, E. Shayesteh and P. Hilber, "Analysing correlated events in power system using fault statistics," in *International Conference on Probabilistic Methods Applied to Power Systems*, pp. 1–6, 2016.

[6] C. N. Barati, S. A. Hosseini, S. Rangan, P. Liu, T. Korakis, S. P. Shivendra and S. R. Theodore, "NTCS: A real time flow-based network traffic classification system," *IEEE Transactions on Wireless Communications*, vol. 14, pp. 6664–6678, 2015.

[7] R. Bardenet, A. Doucet and C. Holmes, "Towards scaling up markov chain monte carlo: An adaptive subsampling approach," in *Proceedings of the 31st International Conference on Machine Learning*, vol. 4, pp. 405–413, 2014.

[8] T. Beery and M. Cherny, *Watching the Watchdog: Protecting Kerberos Authentication with Network Monitoring*, Technical Report, 2015.

[9] Z. Conghua and C. Meiling, "Analysis of fast and secure protocol based on continuous-time markov chain," *Communications, China*, vol. 10, no. 8, pp. 137–149, 2013.

[10] R. Hewett and P. I. Likelihoods, "Exploitability of network vulnerabilities in a large-Scale attack model Kijsanayothin," *International Journal of Network Security*, vol. 17, no. 4, pp. 383–394, 2015.

[11] N. Kamesh and S. Priya, "Security enhancement of authenticated rfid generation," *International Journal of Applied Engineering Research*, vol. 9, no. 22, pp. 5968–5974, 2014.

[12] M. Korczynski, "Classifying application flows and intrusion detection in internet traffic," pp. 1–138, 2012.

[13] M. Korczynski and A. Duda, "Markov chain fingerprinting to classify encrypted traffic," *Proceedings - IEEE INFOCOM*, pp. 781–789, 2014.

[14] S. Liu, A. Takeda, T. Suzuki and K. Fukumizu, "Robust density ratio estimation: Trimming the likelihood ratio," pp. 25, 2017.

[15] A Lowdell and Johan Sakbas, "Achieving authentication and authorization in the combine system," no. 5, pp. 1–25, 2009.

[16] R. Muradore and D. Quaglia, "Energy-efficient intrusion detection and mitigation for networked control systems security," *IEEE Transactions on Industrial Informatics*, vol. 11, no. 3, pp. 830–840, 2015.

[17] T. A. T. Nguyen and T. K. Dang, "Combining fuzzy extractor in biometric-kerberos based authentication protocol," in *Proceedings International Conference on Advanced Computing and Applications*, vol. 23, no. 2, pp. 1–6, 2016.

[18] S. Pan, T. Morris, S. Member, U. Adhikari and S. Member, "Developing a hybrid intrusion detection system using data mining for power systems," *IEEE*

Transactions on Smart Grid, vol. 6, no. 6, pp. 3104–3113, 2015.

[19] S. S. L. Pereira, J. L. C. Silva and J. E. B. Maia, "Ntcs: A real time flow-based network traffic classification system," in *Proceedings of the 10th International Conference on Network and Service Management*, vol. 13, pp. 368–371, 2015.

[20] E. Popoola and A. Adewumi, "Efficient feature selection technique for network intrusion detection system using discrete differential evolution and decision tree," *International Journal of Network Security*, vol. 19, no. 5, pp. 660–669, 2017.

[21] Q. S. Qassim, A. M. Zin and M. J. A. Aziz, "Anomalies classification approach for network-based intrusion detection system," *International Journal of Network Security*, vol. 18, no. 6, pp. 1159–1172, 2016.

[22] K. K. Ravulakollu, "A hybrid intrusion detection system : Integrating hybrid feature selection approach with heterogeneous ensemble of intelligent classifiers," *International Journal of Network Security*, vol. 20, no. 1, pp. 40–53, 2018.

[23] M. Shen, M. Wei, L. Zhu and M. Wang, "Classification of encrypted traffic with second-order markov chains and application attribute bigrams," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 8, pp. 1830–1843, 2017.

[24] W. Stallings, *Computer and Data Communications Technology*, 2011.

[25] M. K. Steven, *Fundamentals of Statistical Signal Processing Detection Theory*, vol. 2, Prentic Hall PTR, 1998.

[26] Z. Tbatou, A. Asimi, Y. Asimi, Y. Sadqi and A. Guezzaz, "A new mutuel kerberos authentication protocol for distributed systems," *International Journal of Network Security*, vol. 19, no. 6, pp. 889–898, 2017.

[27] N. Ye, "A markov chain model of temporal behavior for anomaly detection," *Information Assurance and Security*, no. 4, pp. 171–174, 2000.

[28] N. Ye, X. Li, Q. Chen, S. M. Emran and M. Xu, "Probabilistic techniques for intrusion detection based on computer audit data," *IEEE Transactions on Systems, Man and Cybernetics Part A:Systems and Humans*, vol. 31, no. 4, pp. 266–274, 2001.

[29] N. Ye, S. Member, Y. Zhang and C. M. Borror, "Robustness of the markov-chain model for cyber-attack detection us air force office of scientific research," *IEEE Transactions on Reliability*, vol. 53, no. 1, pp. 116–123, 2004.

[30] X. You and L. Zhang, "Improved authentication model based on kerberos protocol," *Advances in Intelligent and Soft Computing*, vol. 128, no. 2, pp. 593–599, 2011.

[31] C. Zhou, S. Huang, N. Xiong, S.-H. Yang, H. Li, Y. Qin and X. Li, "Design and analysis of multimodel-based anomaly intrusion detection systems in industrial process automation," *IEEE Transactions on Systems, Man and Cybernetics: Systems*, vol. 45, no. 10, pp. 1345–1360, 2015.

Biography

Fadi Al-Ayed received his B.S. degree in 2010 and his M.S. degree in 2012, both in Computer Science from Arizona State University, USA. Currently, he is pursuing his Ph.D. degree in Computer Science at The Catholic University of America. His current research interests include Compute Network and Security, System Security, Data Retrieval and Data Mining.

Chunqiang Hu received his M.S degree and first PhD degree in computer Science and Technology from Chongqing University, Chongqing, China, in 2009 and 2013, respectively. He obtained his B.S. degree in Computer Science and Technology from Southwest University, Chongqing, China, in 2006. He was a visiting scholar at The George Washington University from Jan., 2011 to Dec., 2011, and then got his second PhD degree in Computer Science, The George Washington University, Washington DC in 2016. He won the Best Paper Award in ACM PAMCO 2016. He is a postdoctoral researcher in The Catholic University of America. His current research interests include privacy-aware computing, big data security and privacy, wireless and mobile security, applied cryptography, and algorithm design and analysis. He is a member of IEEE and ACM.

Hang Liu joined the Catholic University of America as an Associate Professor in the Department of Electrical Engineering and Computer Science in 2013. Prior to joining CUA, he had more than 10 years of research experience in networking industry, and worked in senior research and management positions at several companies. He also led several industry-university collaborative research projects. He was an adjunct professor of WINLAB, ECE Dept., Rutgers University from 2004 to 2012. Dr. Liu was an active participant in the IEEE 802 wireless standards and 3GPP standards. He was the editor of the IEEE 802.11aa standard and received an IEEE recognition award. He was also the rapporteur of a 3GPP work item. Hang Liu received a Ph.D. degree in Electrical Engineering from the University of Pennsylvania. His research interests include wireless communications and networking, Internet of Things, mobile computing, future Internet architecture and protocols, content distribution, video streaming, and network security.

Research on Cloud Computing Security Risk Assessment Based on Information Entropy and Markov Chain

Ming Yang[1], Rong Jiang[1], Tilei Gao[1], Wanyu Xie[2] and Jia Wang[1]
(Corresponding author: Rong Jiang)
School of Information, Yunnan University of Finance and Economics, Kunming 650221, China[1]
2502 Gaodeng St, Chenggong Qu, Kunming Shi, Yunnan Sheng, China
Personnel Department, KunMing Metallurgy College, Kunming 650033, China[2]
(Email: jiangrong@ynu.edu.cn)
(Received Mar. 31, 2017; revised and accepted July 13, 2017)

Abstract

The measurement and assessment of risk is an important basis for the research of cloud computing security risk, it can provide important data for risk management decisions. However, due to the uncertainties of risk occurrences and losses, actual risk have multiple stochastic states, make the research of cloud computing risk become more difficult. In order to measure the risk and avoid the influence of subjective factors, a measurement and assessment model of cloud computing risk is established in this paper. The established model used Markov chain to describe random risk environment, and used information entropy to measure risk, effectively reduced the existing subjective factors in the assessment process, provided a practical and reliable method for risk management decisions.

Keywords: Cloud Computing Security; Information Entropy; Markov Chain; Risk Assessment; Risk Measurement

1 Introduction

While providing users with strong computing power and huge application resource, cloud computing also brought potential security threats to users. According to the Global survey results of Gartner, IDC and Unisys [1, 18, 23, 24], security problems have become an important factor for users while selecting cloud computing services.

Cloud computing security is threatened by many factors. These factors are not only technical defects, but also include non-technical factors, such as the lack of management, limitations of laws and the problem of geographically distribution, which bring challenges to cloud computing risk management decisions [19]. In risk management decisions, due to uncertainties the occurrences of cloud computing risk have a variety of random state. Therefore, how to effectively measure and assess the actual risk has become the key to risk management decisions.

Based on the viewpoints proposed in the report "Assessing the security risks of cloud computing" [11], this paper stands on the perspective of cloud computing service providers, and refers to the cloud computing security risk factors proposed in related literature, establishes a cloud computing security risk attribute hierarchies. And on the basis of the attribute hierarchies, this paper conducts quantitative researches on risk uncertainty with the theory of information entropy and Markov chain, and puts forward a measurement and assessment model for cloud computing security risks. This model proposes a risk measurement method, and establishes a risk assessment hierarchy, which solves the problem in measuring abstract risk. Finally, a case was conducted, which shows that the established model can be used to measure objectively the existing risks in a real process and has an important reference value for the future development of cloud computing.

The organization of this paper is as follows: Section 1: Introduction. Introduce the research contents and significance of this paper. Section 2: Related researches. This chapter discusses current research situation about risk factors, risk measurement and risk assessment, and put forward the problems that need to be solved. Section 3: Cloud computing risk and information entropy. This chapter proposes a concept of cloud computing risk entropy, establishes an attribute hierarchy of cloud computing security risk, and describes the cloud computing risk environment with Markov chain. Section 4: The measurement and assessment model of cloud computing risk. On the basis of above research results, this chapter proposes a measurement and assessment model of cloud computing risk, and gives the calculation steps. Section 5: Case analysis. This chapter makes a case research on the

cloud computing security risk of a firm's e-commerce platform with the established model in Section 4. Section 6: Conclusion. Summarize the related research in this paper and point out the future research directions.

2 Related Researches

Cloud service involves many characters, and contains complex information. So while researching cloud computing security risks, it firstly requires organizing the risk factors, and sorting out the logical relationship between them.

2.1 Risk Factors

The report [11] published in Gartner refers to that the risk assessment of cloud computing should be carried out from the data safety, legal risk, investigation support and the survival ability of service providers *etc.* ENISA [3] emphasizes the cloud computing security weakness lies in the defect of management and the lack of laws compliance. Deng [5] analyzes the security problem of cloud computing from different service level based on hadoop, and finds that these security problem mainly include physical infrastructure security, data security, application security, interface security, user rights management security and legal risk *etc.* Cheng [4] takes the information security risks as evaluation target, and establishes an assessment index system which has 35 risk factors, and proposed an new assessment method for the cloud service information security based on AHP (analytic hierarchy process) method; Jiang [13] on the basis of the risk security protection requirements in China, divides the cloud computing security into five aspects as physical security, network security, host system security, application security and data security respectively, and uses AHP method to assign weight for each index, finally puts forward the cloud computing security evaluation model based on the risk security protection. Feng [9] mentioned that the focus of cloud computing security are laws and regulations, business risk management, authentication and access control, application security and physical security.

The above literatures discuss the risk factors of cloud computing security from different aspects, and make a quantitative analysis, provide an important reference value for this paper research. However, when researching the relationship between each risk, these literatures usually divide the risks into several independent categories which do not overlap, and neglect the uncertainty between each risk, which leads to differences between the research results and the real situation.

2.2 Risk Measurement

To assess the risk, and identify the risk factors, the risk measurement is essential. Such as risk value model VaR (Value at Risk) [27,28], actuarial model [8], coherent risk measurement [10,17], risk matrix analysis method [7] and so on. These models provide important reference value for the current research of risk measurement, but inevitably be influenced by subjective bias.

2.3 The Random State of Risk

As known the cloud computing security risk is independent of each other, when a risk is occurring, it may make other risks appearing together, or it may occur alone, there are a variety of possible states about risk occurrences. So when assessing the risk, all possible states about these risk occurrences are required to consider seriously. But the traditional researches mainly carry on a research on a single risk or similar risks [12, 16, 25], and lack of the comparative analysis about different categories of risk.

2.4 Risk Assessment

In addition, in the risk weighting process, most of the literatures haven't make quantitative analysis on the uncertainty and loss degree of each risk, and haven't established the risk assessment system. These research [2, 14, 15, 20, 21, 26] results often focus on the technical risk, not to the other risk factors, and therefore can't give a comprehensive comparison for all kinds of risks from different levels and dimensions.

The above problems all need to be solved in the process of cloud computing risk assessment, and are also the main research content of this paper. Therefore, the first for this article to do is sorting out the risk factors. On the basis of the risk factors, this paper will establish a risk attribute hierarchies with cross relation by using Markov chain to simulate the actual cloud computing risk environment, and carry on a quantitative analysis around the uncertainties of risk occurrences and losses, so that to realizes the quantitative risk analysis from different levels and angles.

3 Cloud Computing Risk and Information Entropy

3.1 Cloud Computing Risk Entropy

Due to the characteristics of cloud computing service itself, the probability of risk occurrence P(x), the risk loss C(x), and the possible occurrence states of risk environments are all uncertain. Therefore, considering the uncertainty of risks, this paper wants to use information entropy method to measure the size of cloud computing risk.

Information refers to the reduction of uncertainty in course of people cognition, in order to quantitatively describe the degree of information uncertainty, the theory founder Shannon proposed the concept of information entropy, and used it to describe the size of information contained in system.

Suppose that a research object X contains n possible result X_i, $X = \{X_1 X_2, \ldots, X_n\}$, in which the occurrence probability of each result is $P(X_i) \sum P(X_i) = 1$, thus the information entropy of this object is $H(X) = -\sum_{i=1}^{n} P(X_i) \log_2 P(X_i)$. Its value is bigger, means that more information of the object contains, more complex the object is, and more high the uncertainty degree is.

When the object has only one possible outcome, now $P(X_1) = 1$, its information entropy $H(X) = 0$, means that the object does not exist any uncertain information; On the contrary, when the object contains N possible outcome, and the occurrence probability of each result is equal as $P(X_1) = P(X_2) =, \ldots, = P(X_n)$, its information entropy will reached a maximum value as $H(X) = \log_2 n$, means that the object reaches the highest uncertainty degree.

However, in the actual situation, information entropy is almost impossible to reach maximum or minimum, and it usually located a value between maximum and minimum.

According to the above theorem, when there is only one possible risk in process of cloud computing, the goal of risk management and maintenance is clear, the risk will be easier to maintain. Conversely, when there is a variety of possible risk, the risk maintenance will be more uncertain. Therefore, it can use the information entropy to describe the uncertainty degree of cloud computing risk. The higher risk uncertainty degree is, the greater risk entropy is, means the risk will be more difficult to control; on the other hand, the lower risk uncertainty degree is, the clearer that the goal of risk maintain is, and the easier risk will be controlled.

3.2 Cloud Computing Security Risk Attribute Hierarchies

Different from the traditional analysis of cloud computing risk, in order to realize the calculation and analysis on cloud computing security from different levels and angles, this paper divides the cloud computing security risk attribute into three levels, as shown in Figure 1 (Cross analysis of cloud computing risk).

The three layers' meanings are:

Target layer: The goal of this paper research;

Risk class layer: The different classes of cloud computing risk, uses $\beta_i, i = 1, 2, \ldots, n$ to express each risk class;

Risk factor layer: The risk factors influencing the cloud computing security, uses $\alpha_j, i = 1, 2, \ldots, m$ to express each risk factor;

This risk attributes hierarchies is different from the traditional research hierarchies, there is complex cross relationships between risk class layer and risk factor layer, which can better reflect the random environment of cloud computing risk.

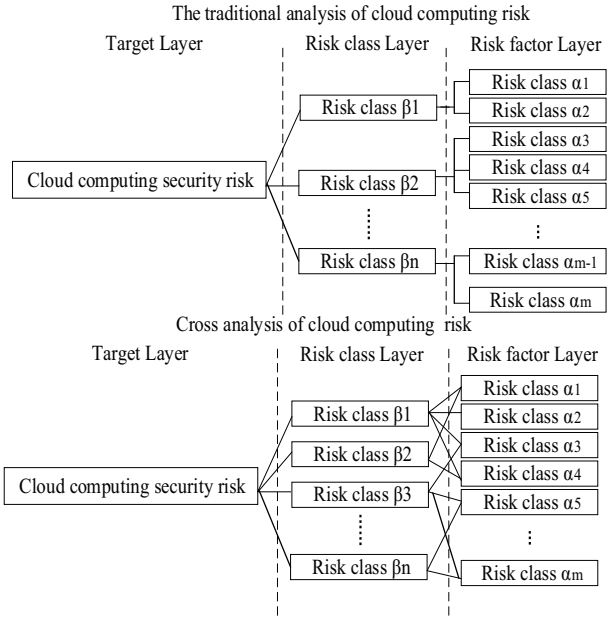


Figure 1: The attribute hierarchies of cloud computing risks

1) The degree of risk uncertainty.
Uses $P(\alpha_j)$ to express the threat frequency of risk factor α_j to cloud security, and uses $P(\beta_i \alpha_j)$ to express the entropy weight of risk factor α_j relative to risk class β_i; Assuming that the class β_i contains K risk factors, thus the calculation formula of $P(\beta_i \alpha_j)$ is as follow:

$$p(\beta_i, \alpha_j) = \frac{1}{\sum_{j=1}^{k} p(\alpha_j)} p(\alpha_j) \tag{1}$$

Then take it into the information entropy formula, as shown below:

$$C(\beta_i) = \sum_{j=1}^{m} p(\beta_i, \alpha_j) C(\alpha_j) \tag{2}$$

$H(\beta_i)(0 \leq H_i \leq 1)$ is risk entropy,it expresses the uncertainty degree of risk class β_i, the higher its value is, the harder the factors causing the risk could be determined, and the harder risk management decisions will be.

2) The degree of risk loss.
In addition to the uncertainties of risk occurrences, in the risk assessment process it also need to consider the degree of risk loss. The calculation formula of risk loss degree is as follow:

$$\begin{aligned} L(\beta_i) &= (L(\beta_1), L(\beta_2), ..., L(\beta_6)) \\ &= (0.392, 0.482, 0.439, 0.476, 0.377, 0.500) \\ L &= 0.451 \end{aligned} \tag{3}$$

In which, $C(\alpha_j)$ expresses risk loss degree of factor α_j, $P(\beta_i, \alpha_j)$ is the entropy weight of factor α_j relative to risk class β_i. As shown in Equation (3),

$C(\beta_i)$expresses risk loss degree of risk class β_{i}. The higher its value is, the greater its impact on cloud security is.

3.3 Markov Chain and Cloud Computing Risk

As shown in Figure 1, in a cloud computing environment there are n risk classes as $\beta_i = 1, 2, \cdots, n$, and each risk class contains a number of risk factors as $\alpha_j, i = 1, 2, \cdots, m$, so the risk occurrence has a variety of random possible states in actual operation process of cloud computing service.

Markov chain has the mathematical definition, it can describe the state of things' random process, with the transfer matrix Q it can calculate the probability of things' random state [6]. Therefore, this paper prepares to use Markov chain to calculate the steady state probability of each risk class during the long operation process of cloud computing service. The first thing is to define the all possible state sets of cloud computing risks, then establish the transfer matrix between them, as shown below:

$$Q = \begin{bmatrix} P(\beta_{11}) & P(\beta_{12}) & P(\beta_{13}) & \ldots & P(\beta_{1n}) \\ P(\beta_{21}) & P(\beta_{22}) & P(\beta_{23}) & \ldots & P(\beta_{2n}) \\ P(\beta_{31}) & P(\beta_{32}) & P(\beta_{33}) & \ldots & P(\beta_{3n}) \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ P(\beta_{n1}) & P(\beta_{n2}) & P(\beta_{n3}) & \ldots & P(\beta_{nn}) \end{bmatrix} \quad (4)$$

The matrix Q expresses the all possible states of each risk class in cloud computing environment. Among them, diagonal elements $\mathrm{P}(\beta_{\mathrm{ij}})(\mathrm{i} = \mathrm{j})$ represent the probability of each risk class happen alone. Thus $\mathrm{P}(\beta_{\mathrm{ij}})(\mathrm{i} \neq \mathrm{j})$ represent the probability of risk class β_{i} and β_{j} happen at the same time, $\sum_{\mathrm{j}=1}^{\mathrm{n}} \mathrm{P}(\beta_{\mathrm{ij}}) = 1$.

Assuming that the probability of each risk class in the steady state is $\mathrm{P}(\beta_{\mathrm{i}}) = (\mathrm{P}(\beta_1), \mathrm{P}(\beta_2), \ldots, \mathrm{P}(\beta_{\mathrm{n}}))$ $\sum \mathrm{P}(\beta_{\mathrm{i}}) = 1$, it satisfying the following equations:

$$\begin{cases} P(\beta_1) = P(\beta_{11})P(\beta_1) + P(\beta_{12})P(\beta_2) + \cdots \\ \qquad + P(\beta_{1n})P(\beta_n) \\ P(\beta_2) = P(\beta_{21})P(\beta_1) + P(\beta_{22})P(\beta_2) + \cdots \\ \qquad + P(\beta_{2n})P(\beta_n) \\ \qquad \cdots . \\ P(\beta_n) = P(\beta_{n1})P(\beta_1) + P(\beta_{n2})P(\beta_2) + \cdots \\ \qquad + P(\beta_{nn})P(\beta_n) \end{cases} \quad (5)$$

Through solving the equations, it can be obtained the steady-state probability $\mathrm{P}(\beta_{\mathrm{i}}), \mathrm{i} = 1, 2, \ldots, \mathrm{n}$. The higher its value is, the easier this risk class will occur in the steady-state, the greater its threat frequency to cloud security is.

4 The Measurement and Assessment Model of Cloud Computing Risk

4.1 The Assessment System of Cloud Computing Risk

This paper on the basis of the index system proposed by GB/T 22239-2008 [22], refers to the risk factors listed in the report "Assessing the security risks of cloud computing" [11] and the risk assessment index proposed by Cheng [27] and Zhu [29], from 6 aspects to establish a hierarchy of cloud computing risk assessment, as shown in Figure 2.

4.2 The Process of Measurement and Assessment Based on Information Entropy

After establishing the risk assessment system, this paper will make detailed measurement and assessment from three aspects: the degree of risk uncertainty, the degree of risk loss and the threat frequency of risk. Its process is as follows:

Step 1: Establish the assessment table as Table 1 and Table 2, and assign weight to the $\mathrm{P}(\alpha_{\mathrm{j}})$ and $\mathrm{C}(\alpha_{\mathrm{j}})$ of risk factors in third layer according to the assessments of 15 domain experts.

Table 1: The assessment table of risk frequency$\mathrm{P}(\alpha_{\mathrm{j}})$

Weight	Level	Specific definitions
1	Very high	The frequencies of risk factors are very high, almost inevitable in actual situation
0.8	high	The frequencies of risk factors are high, often occur in most cases
0.6	Medium	The frequencies of risk factors are normal, may occur in some cases
0.4	low	The frequencies of risk factors are low, it will occur in a minority of cases
0.2	Very low	The frequencies of risk factors are very low, almost never happen in a minority of cases

Assuming that experts' assessment distributions of risk frequencies and risk losses are$\mathrm{P}(\mathrm{x}, \mathrm{y})$ and $\mathrm{C}(\mathrm{x}, \mathrm{y})$, in which x expresses risk factors and y expresses the weight level. Thus calculations of $\mathrm{P}(\alpha_{\mathrm{j}})$ and $\mathrm{C}(\alpha_{\mathrm{j}})$

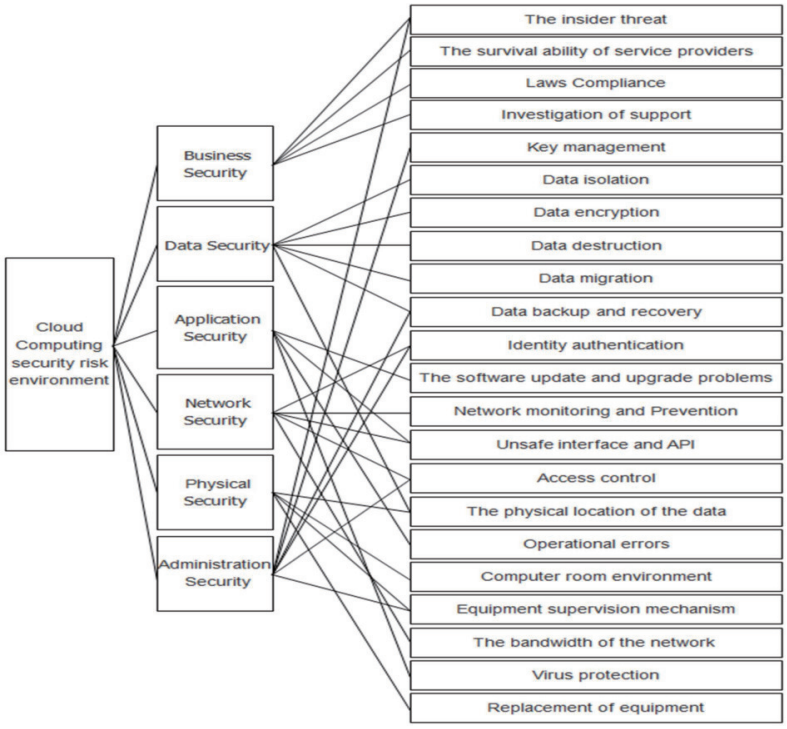


Figure 2: The assessment hierarchy of cloud computing security risk

are shown as the following formula:

$$P(\alpha_j) = (0.2, 0.4, 0.6, 0.8, 1)(p(x,1), p(x,2), \cdots, p(x,5))$$
$$C(\alpha_j) = (0.2, 0.4, 0.6, 0.8, 1)(c(x,1), c(x,2), \cdots, c(x,5)) \quad (6)$$

The $P(\alpha_j)$ and $C(\alpha_j)$ depend on *experts'* assessment distribution, the more dispersed expert assessments are, the higher assessment results' uncertainties are. Conversely, the more concentrated expert assessments are, the higher assessment results' certainties are, so the assessment weight of each risk factor can be defined as the following formula:

$$V(\alpha_j) = \sqrt[2]{(1 - \sum_{j=1}^{5} p_{ij} \log_5 p_{ij})(1 - \sum_{j=1}^{5} c_{ij} \log_5 c_{ij})} \quad (7)$$

The value of $V(\alpha_j)$ expresses its contribution on risk assessment, the higher its value is, the greater its contribution is.

Step 2: According to the classification in Figure 2, use Equation (1) to calculate the entropy weight coefficient $P(\beta_i, \alpha_j)$;

Step 3: Put the $P(\beta_i, \alpha_j)$into Equations (2) and (3), and calculate the degree of risk uncertainty $H(\beta_i)$ and the degree of risk losses $C(\beta_i)$.

Step 4: According to Markov chain principle, calculate the steady-state probability of each risk class $P(\beta_i) = (P(\beta_1), P(\beta_2), \ldots, P(\beta_6))$.

Firstly, according to the assessment system of cloud computing security risk shown in Fig.2, and combined with the frequency $P(\alpha_j)$ of each risk factor to establish the transfer matrix between each risk class, as follows:

$$Q = \begin{bmatrix} P(\beta_{11}) & P(\beta_{12}) & P(\beta_{13}) & \cdots & P(\beta_{16}) \\ P(\beta_{21}) & P(\beta_{22}) & P(\beta_{23}) & \cdots & P(\beta_{26}) \\ P(\beta_{31}) & P(\beta_{32}) & P(\beta_{33}) & \cdots & P(\beta_{36}) \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ P(\beta_{61}) & P(\beta_{62}) & P(\beta_{63}) & \cdots & P(\beta_{66}) \end{bmatrix} \quad (8)$$

In the matrix, the diagonal elements $P(\beta_{ii})$ represent the probability of the risk class β_i occurred alone, and the elements $P(\beta_{ij})$ represent the probabilities of risk class β_i and β_j happen at the same time, its value depends on the factors contained in each risk class.

As shown in the following example. The Markov transition matrix of them is as follows:

$$\begin{bmatrix} P(\beta_{11}) & P(\beta_{12}) \\ P(\beta_{21}) & P(\beta_{22}) \end{bmatrix}$$
$$= \begin{bmatrix} \frac{1}{\sum_{i=1}^{3} P(\alpha_i)} P(\alpha_1) + P(\alpha_2) & \frac{1}{\sum_{i=3}^{5} P(\alpha_i)} P(\alpha_3) \\ \frac{1}{\sum_{i=1}^{3} P(\alpha_i)} P(\alpha_3) & \frac{1}{\sum_{i=3}^{5} P(\alpha_i)} P(\alpha_4) + P(\alpha_5) \end{bmatrix}$$

After establishing the Markov transition matrix, suppose that the steady-state probability of each risk class in second layer is $P(\beta_i) = (P(\beta_1), P(\beta_2), \ldots,$

Table 2: The assessment table of risk loss degree C(α_j)

Weight	Level	Specific definitions
1	Very high	Once the risk occurs will cause devastating losses
0.8	high	The impact of risk is larger, maintenance needs higher funds
0.6	Medium	The impact and economic loss caused by risk is normal
0.4	low	The impact caused by risk is lower, and the maintenance funds required lower
0.2	Very low	The impact caused by risk can be ignored, and hardly need maintenance

Table 3: Two different risk classes

	risk class β_1	**risk class β_2**
risk factors	α_1,α_2,α_3	α_3,α_4,α_5

$P(\beta_6))\sum \mathrm{P}(\beta_i)=1$, then put it into Equation (5) to calculate the steady-state probability.

Step 5: Define the grade of cloud computing security risk, and make integrated risk assessment.

The definition of cloud computing security risk grade contains three factors: the degree of risk uncertainty H(β_i), the degree of risk loss C(β_i) and the frequency of risk occurrence P(β_i). The specific definitions are as shown in Table 4.

The calculation formula of the grade of each risk class is as follows:

$$L(\beta_i) = \sqrt[3]{H(\beta_i)C(\beta_i)P(\beta_i)} \tag{9}$$

According to the definition in the Table 4, the greater value of L(β_i) is, the higher occurrence frequency of this risk class is, the harder risk maintenance is, and the greater risk loss is.

Next, on the basis of L(β_i), this paper will further assess the whole cloud computing security risk grade, its calculation formula is as follows:

$$L = (L(\beta_1), L(\beta_2), \cdots, L(\beta_6))(V(\beta_1), V(\beta_2), \cdots, V(\beta_6))^T \tag{10}$$

Among them,V (β_i) , i = 1, 2, . . . , 6 expresses the assessment weight of each risk class β_i, its calculation formula is as follows:

$$V(\beta_i) = \frac{1}{\sum_{i=1}^{6}\sum_{j=1}^{m} V(\alpha_j)} \sum_{j=1}^{m} V(\alpha_j) \tag{11}$$

Among them, m is the counts of risk factors contained in risk class β_i. The value of V (β_i) expresses its impact on the entire cloud security.

Table 4: The grade of cloud computing security risk

Grade	Specific definitions
$0.8 < L < 1$	The factors causing risk can't be determined. Once risks occur, cloud service will be almost impossible to maintain success. Its cloud security belongs the catastrophic risk
$0.6 < L \leq 0.8$	The factors causing risk are many and be difficult to determine. Once risks occur, they will directly affect the normal operation process of cloud services
$0.4 < L \leq 0.6$	There will be some impact on the operation process of cloud services. The cloud security belongs the general risk level, its service need maintenance routine,
$0.2 < L \leq 0.4$	Risk maintenance goals is clear, its cloud computing services are well-managed
$0 < L \leq 0.2$	Risk maintenance goal was very clear, there is almost not any impact on cloud computing services,the risk impact often can be ignored

5 Case Analysis

5.1 The Process of Calculation

According to the risk assessment system established in this paper, this article makes a case research on the cloud computing security risk of a firm's e-commerce platform.

Step 1: Trough the experts scoring, the assessment distribution results are shown in Table 5.

Step 2: Make normalization processing, get the entropy weight coefficient of $P(\beta_i, \alpha_j)$,as shown in Table 6.

Step 3: According to Formula (2) and (3), calculate the degree of risk uncertainty $H(\beta_i)$ and the degree of risk loss (β_i), get the results as follows:

$$\begin{aligned} H(\beta_i) &= (H(\beta_1), H(\beta_2), \cdots, H(\beta_6)) \\ &= (0.941, 0.978, 0.992, 0.993, 0.992, 0.987) \\ C(\beta_i) &= (C(\beta_1), C(\beta_2), \cdots, C(\beta_6)) \\ &= (0.622, 0.594, 0.474, 0.500, 0.562, 0.592) \end{aligned}$$

Step 4: According to the principle of Markov chain, establish the Markov transfer matrix of each risk class.

$$\begin{vmatrix} 0.574 & 0.000 & 0.000 & 0.000 & 0.000 & 0.426 \\ 0.000 & 0.758 & 0.000 & 0.000 & 0.116 & 0.126 \\ 0.000 & 0.000 & 0.761 & 0.239 & 0.000 & 0.000 \\ 0.000 & 0.000 & 0.198 & 0.436 & 0.000 & 0.366 \\ 0.000 & 0.234 & 0.000 & 0.000 & 0.486 & 0.280 \end{vmatrix}$$

Table 5: The results of assessment distribution

	assessment distribution of **P**(**xy**)						assessment distribution of **C**(**xy**)					
Risk factors α_j	**0.2**	**0.4**	**0.6**	**0.8**	**1**	**P**(α_j)	**0.2**	**0.4**	**0.6**	**0.8**	**1**	**C**(α_j)
Identity authentication	0.00	0.27	0.60	0.13	0.00	0.573	0.00	0.07	0.73	0.20	0.00	0.627
Access control	0.00	0.07	0.80	0.13	0.00	0.613	0.00	0.07	0.87	0.07	0.00	0.600
Laws Compliance	0.27	0.73	0.00	0.00	0.00	0.347	0.07	0.33	0.40	0.20	0.00	0.547
Investigation of support	0.47	0.53	0.00	0.00	0.00	0.307	0.27	0.60	0.13	0.00	0.00	0.373
Key management	0.00	0.20	0.67	0.13	0.00	0.587	0.00	0.33	0.47	0.20	0.00	0.573
Data isolation	0.00	0.13	0.53	0.33	0.00	0.640	0.00	0.27	0.60	0.13	0.00	0.573
Data encryption	0.00	0.13	0.40	0.33	0.13	0.693	0.00	0.00	0.47	0.47	0.07	0.720
Data destruction	0.07	0.80	0.13	0.00	0.00	0.413	0.07	0.27	0.47	0.20	0.00	0.560
Data migration	0.07	0.73	0.20	0.00	0.00	0.427	0.00	0.20	0.80	0.00	0.00	0.560
Data backup and recovery	0.20	0.80	0.00	0.00	0.00	0.360	0.07	0.53	0.20	0.20	0.00	0.507
The insider threat	0.00	0.13	0.53	0.27	0.07	0.653	0.00	0.07	0.47	0.33	0.13	0.707
software update problems	0.00	0.00	0.53	0.27	0.20	0.733	0.13	0.60	0.27	0.00	0.00	0.427
Network monitoring and Prevention	0.00	0.20	0.73	0.07	0.00	0.573	0.20	0.40	0.40	0.00	0.00	0.440
Unsafe interface and API	0.00	0.07	0.67	0.27	0.00	0.640	0.00	0.07	0.60	0.33	0.00	0.653
survival ability of service providers	0.87	0.13	0.00	0.00	0.00	0.227	0.00	0.00	0.07	0.73	0.20	0.827
data physical location	0.33	0.67	0.00	0.00	0.00	0.333	0.13	0.13	0.53	0.20	0.00	0.560
Operational errors	0.00	0.00	0.33	0.40	0.27	0.787	0.13	0.67	0.20	0.00	0.00	0.413
Computer room environment	0.20	0.80	0.00	0.00	0.00	0.360	0.00	0.00	0.33	0.53	0.13	0.760
Equipment supervision mechanism	0.00	1.00	0.00	0.00	0.00	0.400	0.13	0.53	0.27	0.07	0.00	0.453
bandwidth of network	0.00	0.00	0.13	0.53	0.33	0.840	0.73	0.20	0.07	0.00	0.00	0.267
Virus protection	0.00	0.40	0.60	0.00	0.00	0.520	0.07	0.80	0.13	0.00	0.00	0.413
Replacement of equipment	0.33	0.67	0.00	0.00	0.00	0.333	0.00	0.40	0.53	0.07	0.00	0.533

Table 6: The entropy weight coefficient P($\beta_i\alpha_j$) of each risk class

Business Security β_1	**P**(α_j)	**P**($\beta_1\alpha_j$)	**Data Security** β_2	**P**(α_j)	**P**($\beta_2\alpha_j$)
The insider threat	0.653	0.426	data physical location	0.333	0.116
survival ability	0.227	0.148	Data encryption	0.693	0.242
Laws Compliance	0.347	0.226	Data backup and data recovery	0.360	0.126
investigation support	0.307	0.200	Data isolation	0.640	0.223
			Data destruction	0.413	0.144
			Data migration	0.427	0.149
Application Security β_3	**P**(α_j)	**P**($\beta_3\alpha_j$)	**Network Security** β_4	**P**(α_j)	**P**($\beta_4\alpha_j$)
Virus protection	0.520	0.194	bandwidth of the network	0.840	0.259
Operational errors	0.787	0.294	Network monitoring and Prevention	0.573	0.177
Unsafe interface	0.640	0.239	Unsafe interface	0.640	0.198
problems of software update	0.733	0.274	Identity authentication	0.573	0.177
			Access control	0.613	0.189
Physical Security β_5	**P**(α_j)	**P**($\beta_5\alpha_j$)	**Administration Security** β_6	**P**(α_j)	**P**($\beta_6\alpha_j$)
data physical location	0.333	0.217	Data backup and recovery	0.360	0.113
Equipment supervision mechanism	0.400	0.280	Equipment supervision mechanism	0.400	0.126
Computer room environment	0.360	0.252	Identity authentication	0.573	0.180
Replacement of equipment	0.333	0.234	Access control	0.613	0.192
			The insider threat	0.653	0.205
			Key management	0.587	0.184

Put the above data into Equation (5) to calculate, it can get the steady-state probability of each risk class in the long-term operation process of cloud computing service, as shown below:

$$p(\beta_i) = (p(\beta_1), p(\beta_2), \cdots, p(\beta_6)) = (0.103, 0.192, 0.179, 0.217, 0.096, 0.213)$$

Step 5: According to Equations (9) and (10), calculate the risk grade of each class and the risk grade of the whole environments, get the results shown below:

$$L(\beta_i) = (L(\beta_1), L(\beta_2), \cdots, L(\beta_6)) = (0.392, 0.482, 0.439, 0.476, 0.377, 0.500)$$
$$L = 0.451$$

5.2 Analysis of Research Results

The model presented in this paper realizes measurement and assessment of cloud computing security from different layers, different angles and different classes. The above research results are summarized, as shown in Table 7.

Table 7: The research results of risk measurement and assessment

	β_1	β_2	β_3	β_4	β_5	β_6
$\mathbf{H}(\beta_i)$	0.941	0.978	0.992	0.993	0.992	0.987
$\mathbf{C}(\beta_i)$	0.622	0.594	0.474	0.500	0.562	0.592
$\mathbf{P}(\beta_i)$	0.103	0.192	0.179	0.217	0.096	0.213
$\mathbf{L}(\beta_i)$	0.392	0.482	0.439	0.476	0.377	0.500
the risk grade of entire cloud computing security						
$\boldsymbol{L}$	0.451					

Through the analysis of the research results, it can be found:

1) $L = 0.451$, it expresses the grade of whole cloud computing security risk. This value illustrates that this firm's cloud computing security belongs the general risk level, its cloud computing service exists some risk, need maintenance routine, and its service is in the acceptable level.

2) $L(\beta_6) = 0.5, L(\beta_2) = 0.482$ and $L(\beta_4) = 0.476$, these values are more higher than other risk grade of the whole system. These data illustrate that the administration Security, data security and network security are the most threats to this firm's cloud security, which are the key to decide the security of this e-commerce platform and should be paid more attention in the risk management decisions. Conversely, $L(\beta_1) = 0.392$ and $L(\beta_5) = 0.377$ means that this firm's physical Security and business Security is well-managed.

In addition, according to the data of $P(\beta_i)$, $C(\beta_i)$ and $H(\beta_i)$, it can be found:

1) $P(\beta_4)$=*0.217*, it means that the occurrence frequency of network risk is the highest in long-term operation process. If this firm want to improve its cloud security, it should strengthen the protection of network.

2) $C(\beta_1)$=*0.622*, $C(\beta_2)$=*0.594* and $C(\beta_1)$=*0.592*, these data mean that the business risk, data risk and administration risk are the greatest potential threat to this firm's cloud security, once the risks occur they will cause huge losses to the company.

3) Comparing the risk uncertainty, it can be found that only business risk and data risk are lower. It illustrates that only these two risk classes are easier to control compared with the other risk.

On the basis of the above analysis, through the model presented in this paper, it can also make further in-depth analysis around the risk factors in the third layer, so that to provide detailed information for the firm's cloud computing security risk management.

6 Conclusion

This paper, bases on the information entropy theory, makes quantitative research on risk uncertainty, has reduced the influence of subjective factors on the quantitative results, and finally provides a reference standard for risk management decision.

Compared with the past research methods, this paper divides the cloud computing risk into 6 classes and establishes a risk assessment hierarchy with cross relations.

Combined with the Markov chain, this paper, calculates the steady-state probability of each risk class in the stable cloud computing process, makes up the lack of research on the uncertainty between each risk, and gives the definition of risk grade based on information entropy.

In the following work, author will still continue to identify and add new security risk factors of cloud computing, and avoid redundant factors, so that to provide more detailed risk assessment system.

Acknowledgments

This work was supported by National Natural Science Foundation of China (Nos. 61763048, 61263022 and 61303234), National Social Science Foundation of China (No. 12XTQ012), Science and Technology Foundation of Yunan Province (No. 2017FB095), the 18th Yunnan Young and Middle-aged Academic and Technical Leaders Reserve Personnel Training Program (No.2015HB038), Yunnan Province Applied Basic Research Project (No.2016FD060) and Science Research Foundation of Yunnan Provincial Department of Education (No. 2017ZZX001).

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

References

[1] M. Ahmad, *Security Risks of Cloud Computing and Its Emergence as 5th Utility Service*. Springer Berlin Heidelberg, 2010.

[2] Y. Cai, "Security risk assessment model of information system based on cloud computing," *China Management Informationization*, vol. 12, pp. 75–77, 2010.

[3] D. Catteddu, "Cloud computing: benefits, risks and recommendations for information security," in *Web application security*. Springer, 2010, pp. 17–17.

[4] Y. Cheng, "The evaluation index of cloud service information security and its method study," *Beijing Jiaotong University*, 2013.

[5] Q. Deng, "Research on the security mechanism of cloud computing based on hadoop," *Nanjing: Nanjing University of Posts and Telecommunications*, 2013.

[6] X. Duan, M. X. Huang, B. Wan, and X. Yang, "Research on supply chain partner selection based on markov chain dynamic fuzzy evaluation in cloud computing," *Application Research of Computers*, vol. 31, no. 8, pp. 2403–2406, 2014.

[7] Y. Duan, J. Zhao, J. Chen, and G. Bai, "A risk matrix analysis method based on potential risk influence: A case study on cryogenic liquid hydrogen filling system," *Process Safety & Environmental Protection*, vol. 102, pp. 277–287, 2016.

[8] M. Eling and N. Loperfido, "Data breaches: Goodness of fit, pricing, and risk measurement," *Insurance Mathematics & Economics*, vol. 75, p. 126136, 2017.

[9] D. Feng, M. Zhang, Y. Zhang, and Z. Xu, "Cloud computing security research journal of software," *Journal of software, Computer Science*, vol. 22, no. 1, pp. 71–83, 2011.

[10] H. Föllmer and I. Penner, "Consistent risk measures and a non-linear extension of backwards martingale convergence." *Festschrift Masatoshi Fukushima*, pp. 183–202, 2015.

[11] J. Heiser and M. Nicolett, "Assessing the security risks of cloud computing," 2008.

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

[13] Z. W. Jiang, W. R. Zhao, and Y. Liu, "Model for cloud computing security assessment based on classified protection," *Computer Science*, 2013.

[14] C. Joshi and U. K. Singh, "Information security risk management framework for university computing environment," *International Journal of Network Security*, 2017.

[15] M. Jouini and L. B. A. Rabai, "Comparative study of information security risk assessment models for cloud computing systems ," *Procedia Computer Science*, vol. 83, pp. 1084–1089, 2016.

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

[17] S. Mitra, "Efficient option risk measurement with reduced model risk," *Insurance Mathematics & Economics*, 2016.

[18] R. Morrell and A. Chandrashekar, "Cloud computing: new challenges and opportunities," *Network Security*, vol. 2011, no. 10, pp. 18–19, 2011.

[19] A. Mosa, H. M. El-Bakry, S. M. Abd El-Razek, S. Q. Hasan, "A proposed E-government framework based on cloud service architecture," *International Journal of Electronics and Information Engineering*, vol. 5, no. 2, pp. 93–104, 2016.

[20] D. R. D. Santos, R. Marinho, G. R. Schmitt, C. M. Westphall, and C. B. Westphall, "A framework and risk assessment approaches for risk-based access control in the cloud," *Journal of Network & Computer Applications*, vol. 74, pp. 86–97, 2016.

[21] F. U. Sha, Y. Z. Xiao, and M. H. Liao, "An approach for campus information systems security risk assessment based on fuzzy set and entropy weight," *Information Science*, 2013.

[22] Standardization Administration of the People's Republic of China, *Information Security Technology - Baseline for Classified Protection of Information System Security*, GB/T 22239-2008, 2008.

[23] D. Sun, G. Chang, L. Sun, and X. Wang, "Surveying and analyzing security, privacy and trust issues in cloud computing environments," *Procedia Engineering*, vol. 15, no. 1, pp. 2852–2856, 2011.

[24] S. Tanimoto, M. Hiramoto, M. Iwashita, H. Sato, and A. Kanai, "Risk management on the security problem in cloud computing," in *First Acis/jnu International Conference on Computers, Networks, Systems and Industrial Engineering*, 2011, pp. 147–152.

[25] J. Wang, J. Liu, and H. Zhang, "Access control based resource allocation in cloud computing environment," *International Journal of Network Security*, vol. 19, no. 2, pp. 236–243, 2017.

[26] Z. C. Wang, "Research on information security risk assessment based on cloud computing model," *Netinfo Security*, 2011.

[27] Y. C. Xu, "Research status of risk value model in foreign countries," *Foreign Economics & Management*, vol. 27, no. 6, pp. 44–51, 2005.

[28] Z. Zhang, L. Yang, H. Li, and F. Xiang, "A quantitative and qualitative analysis-based security risk assessment for multimedia social networks," *International Journal of Network Security*, vol. 18, no. 1, pp. 43–51, 2016.

[29] S. C. Zhu, X. U. Yu, and M. Y. Jin, "Cloud computing security risk assessment based on level protection strategy," *Computer Security*, 2013.

Biography

Ming Yang is a lecturer at the school of information, Yunnan University of Finance and Economics, China.

He received his Ph.D. in system analysis and integration from the school of software at Yunnan University. His main research interests include information management and data mining.

Rong Jiang is a professor and Ph. D. supervisor at the school of information, Yunnan University of Finance and Economics, China. He received his Ph.D. in system analysis and integration from the school of software at Yunnan University. He has published more than 30 papers and ten books, and has gotten more than 40 prizes in recent years. His main research interests include cloud computing, big data, software engineering, information management, etc.

Tilei Gao is a lecturer at the school of information, Yunnan University of Finance and Economics. He is also a Ph.D candidate in system analysis and integration at the school of software at Yunnan University. His main research interests include software engineering and information management.

Wanyu Xie is a lecturer at Kunming Metallurgy College. She received Her master's degree in computer science and technology from the school of information science and engineering at Yunnan University. Her main research interests is information management.

Jia Wang is a lecturer at the Yunnan University of Finance and Economics. He received his master's degree in software engineering from the school of software at Yunnan University. His main research interests include embedded system architecture and information management.

Study on Trust Model for Multi-users in Cloud Computing

Xu Wu[1,2]

(Corresponding author: Xu Wu)

School of Computer, Electronics and Information, Guangxi University[1]
No. 100 East Daxue Road, Nanning, Guangxi 530004, China
Key Laboratory of Multimedia Communication and Network Technology, Guangxi University, Nanning, China[2]
(Email: xrdz2005@163.com)

(Received Mar. 12, 2017; revised and accepted June 20, 2017)

Abstract

Although some trust management approaches are proposed for cloud computing, these approaches only deal with single and simple trust relationship, trust algorithms in these models are one dimensional, and can not accurately measure the trust relationship between multi-users. In the paper we present a cloud trust model based on trust level agreement. The proposed method can assist cloud computing entities to make good interaction decisions. The main contribution of this paper is to provide a hierarchical trust modeling method to user and improve his or her security situational awareness in the cloud computing environments. The experimental results show that the proposed method has a higher trust accurate rate and interaction success rate, and it is qualified to prevent malicious entities attacks while maintaining efficiency. Our analysis shows a significant improvement in comparison to traditional trust management technology. Our work appears to be the first attempt to research the multi-entities trust management method in cloud computing.

Keywords: Cloud Computing; Decision-Making; Trust Management

1 Introduction

With the development of virtualization technology, computers have transited from the real to a virtual machine; people begin to pursue lightweight computing service [12]. As P2P network, grid computing, utility computing and a series of distributed computing technology are constantly emerging, and they create a new kind of distributed computing technology, cloud computing. Cloud computing brings a shift from heavy IT infrastructure invest for limited resources that are internally managed and owned by a customer to pay per use for IT infrastructure owned by a cloud computing service provider. There are many benefits to cloud computing: lower overall cost of IT ownership, increased flexibility, fault tolerance, locality flexibility ability, and to respond to new business requirements quickly and efficiently [13]. However, cloud computing in industry is not as popular as it is in the academia at present, the reason is that user distrust cloud computing environment, and they are not willing to put their private information and data in the computer of a third party. Therefore, the problem of trust in cloud computing environment is becoming more and more serious; a lack of trust between cloud customers and providers has hindered the universal acceptance of clouds as an increasingly popular approach for the processing of large data sets and computationally expensive programs [1].

A good solution is to leverage trust management technology to build trust for cloud computing [3]. Trust management is fundamental to identify malicious, selfish and compromised nodes which have been authenticated. It has been widely studied in many network environments such as peer-to-peer networks, grid and pervasive computing and so on. Trust is an important aspect in the design and analysis of secure distribution systems. It is also one of the most important concepts guiding decision-making. Trust is a critical part of the process by which relationships develop. It is a before-security issue in the ad hoc networks. By clarifying the trust relationship, it will be much easier to take proper security measures, and make correct decision on any security issues. Trust modeling is a technical approach to represent trust for digital processing. Recently, trust modeling is paid more and more attention in cloud computing. Although some trust management approaches are proposed for cloud computing, these approaches only deal with single and simple trust relationship, trust algorithms in these models are one dimensional, and can't accurately measure the trust relationship between multi-users. Therefore, in the paper we propose a cloud trust model based on a trust level agreement (TLABCTM). The major contributions of this paper can be summarized as follows:

1) We present a hierarchical trust management framework. In the framework, trust is divided into three layers: cloud service provider trust layer (CSPTL), cloud component trust layer (CCTL) and cloud user trust layer (CUTL).

2) We propose a trust level agreement which includes two types: User Trust Level Agreement and Provider Trust Level Agreement. The trust level agreement classifies the identity of entity and service type. Users can obtain correspond cloud service according to their trust level agreement.

3) Our work appears to be the first attempt to research the multi-entities trust management method in cloud computing. The experimental results show that the proposed method has a higher trust accurate rate and interaction success rate, and it is qualified to prevent malicious entities' attacks while maintaining efficiency.

This paper is organized as follows. Section 2 describes related work. In Section 3, the proposed cloud trust model (TLABCTM) is discussed. Section 4 presents the experimental setup used to test the mechanism along with the results. Finally, we conclude with a summary of our results and directions for new research in Section 5.

2 Related Work

The issue of establishing trust in different environments has been discussed by many authors. Kumar *et al.* [9] present a novel approach to secure the Mobile Ad Hoc Networks. Error correcting codes are used to assign identification to resource constrained mobile nodes. This assignment helps to create centralized environment with subgroups, groups and hierarchies.

Deverajan *et al.* [5] propose a new protocol namely Adaptive Fuzzy DoT Threshold Routing Algorithm (AFTRA), which takes into account the Degree of Trust (DoT), connectivity and the energy levels. AFTRA provides all possible routes from the source to the destination. The best route is selected by considering three aspects hop count, trust values and energy. In addition, Deverajan *et al.* [6] also propose a novel trust based system to detect the intrusive behavior. The entire work of this system can be compartmentalized into three phases. They are Pre-eminent node selection, Inter-cluster trust rate computation, Intra-cluster trust rate computation.

Hwang *et al.* [8] distinguish among different service-level agreements (SLAs) by their variable degree of shared responsibility between cloud providers and users. Critical security issues include data integrity, user confidentiality, and trust among providers, individual users, and user groups. The three most popular cloud service models have varying security demands. The work in [17] is a very recently work on trust management in IoT environments. A trusted service platform is established, which provides trust evaluation based on three trust metrics. These metrics include Reputation, Recommendation, and Knowledge. The idea of the proposed method comes from modeling human trust relationship.

In [16] a distributed reputation based trust management system is presented for hybrid cloud computing system. The mechanism can effectively address strategic feedbacks and mitigate unfairness. The performance of the proposed trust management system has been studied in a simulated environment and due to space limitations this information is not fully provided. In order to solve privacy and security problems in the IaaS service layer, a model of trustworthy cloud computing which provides a closed execution environment for the confidential execution of virtual machines was proposed [20]. This work has shown how the problem can be solved using a Trusted Platform Module.

Zhimin *et al.* [19] propose a collaborative trust model for firewalls in cloud computing. The model has three advantages: 1) it uses different security policies for different domains; 2) it considers the transaction contexts, historic data of entities and their influence in the dynamic measurement of the trust value; and 3) the trust model is compatible with the firewall and does not break its local control policies. Hada *et al.* [7] propose a trust model for cloud architecture which uses mobile agent as security agents to acquire useful information from the virtual machine which the user and service provider can utilize to keep track of privacy of their data and virtual machines.

Edna *et al.* [4] presented an overview of the cloud computing paradigm, as well as its main features, architectures and deployment models. Moreover, they identified the main issues related to trust and security in cloud computing environments. Cloud service providers (CSP) should guarantee the services they offer, without violating users' privacy and confidentiality rights. Li *et al.* [11] introduced a multitenancy trusted computing environment model (MTCEM). This model was designed for the IaaS layer with the goal of ensuring a trustworthy cloud computing environment to users.

Pawar *et al.* [15] propose an uncertainty model and define an approach to compute opinion for cloud service providers. Using subjective logic operators along with the computed opinion values, they propose mechanisms to calculate the reputation of cloud service providers. They also evaluate and compare the proposed model with existing reputation models. T-broker [10] presents, a trust-aware service brokering scheme for efficient matching cloud services (or resources) to satisfy various user requests. The experimental results show that, compared with the existing approaches, T-broker yields very good results in many typical cases, and the proposed system is robust to deal with various numbers of dynamic service behavior from multiple cloud sites.

In the paper [14], the author describes the design and implementation of CloudArmor, a reputation-based trust management framework that provides a set of functionalities to deliver Trust as a Service (TaaS). The experimental

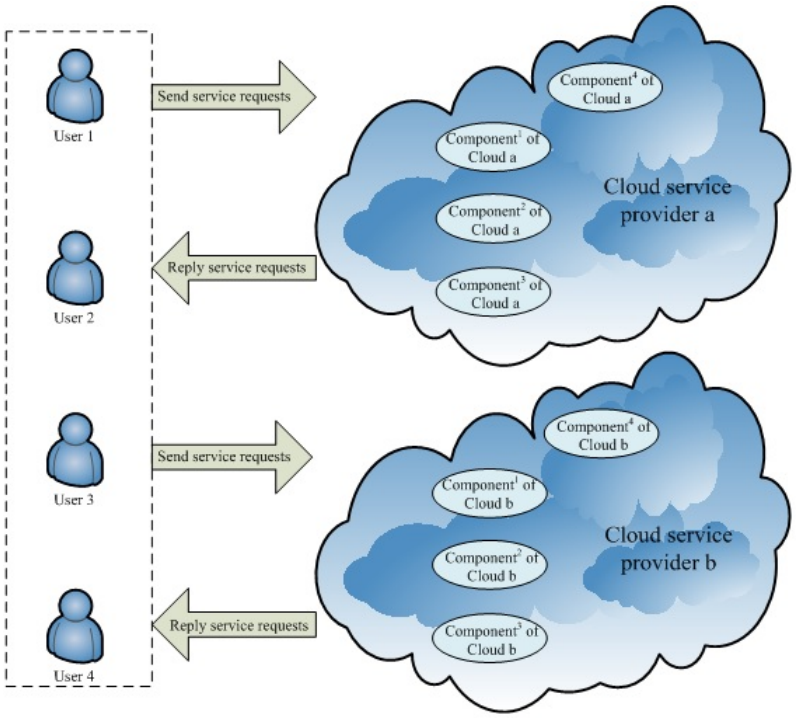


Figure 1: An example of cloud computing environment

results demonstrate the applicability of our approach and show the capability of detecting such malicious behaviors.

3 Cloud Trust Model Based on Trust Level Agreement (TLABCTM)

In this section, we firstly present an example of cloud computing environment. Secondly, we present the definitions about trust level agreement and the main idea of TLABCTM. Finally, we present the details about how to evaluate the trustworthiness of entities.

3.1 Scenario

Cloud computing is a large-scale and dynamic computing environment, so the types of entity which is involved in are different. The current academic circles widely divide these entities into two categories in the cloud computing: cloud service providers and users. However, with the development of cloud computing technology, in cloud computing appears a new entity identity-"component". In Figure 1, every cloud has different components which supply corresponding cloud service to users. The component may be an artificial agent or middleware software. Different entities are linked to Internet in cloud dynamically. These entities include cloud service providers, cloud users and cloud's components as shown in Figure 1, and all entities would dynamically enter or exit the virtual organization in cloud computing environment.

Cloud Service provider (CSP), it provides various cloud service for cloud computing environments, such as Software as a Service, Platform as a Service, Infrastructure as a Service, etc., it is a collection of components.

Cloud User (CU), it is a service entity which uses computing resources, storage resources in Cloud computing environments.

Cloud Component (CC), it is the actual carrier of service in cloud computing environment. Each CSP has j cloud components (*i.e.*, $CC = CC_1, CC_2, \cdots, CC_j$).

A CU sends service request to a CSP. Then the CSP will decide whether the CU is trusted or not. If the CU is trusted, CSP will select certain trusted CC (*i.e.*, component with trust values exceeding a threshold) to supply service.

In Figure 1, there exist multi-trust relationship (trust between CSP and CC, trust between CU and CC, trust between CSP and CU, trust between CU and CU, trust between CSP and CSP, and CC and CC).

3.2 Main idea of TLABCTM

In order to meet multi-entities' trust requirements, we propose a trust level agreement based cloud trust model (TLABCTM). The Figure 2 presents the main idea.

A hierarchical trust management framework is established. In the framework, trust is divided into three layers: cloud service provider trust layer (CSPTL), cloud component trust layer (CCTL) and cloud user trust layer (CUTL). Entities are divided into three types in TLABCTM: CSP, CU and CC. A cross-layer trust flow is

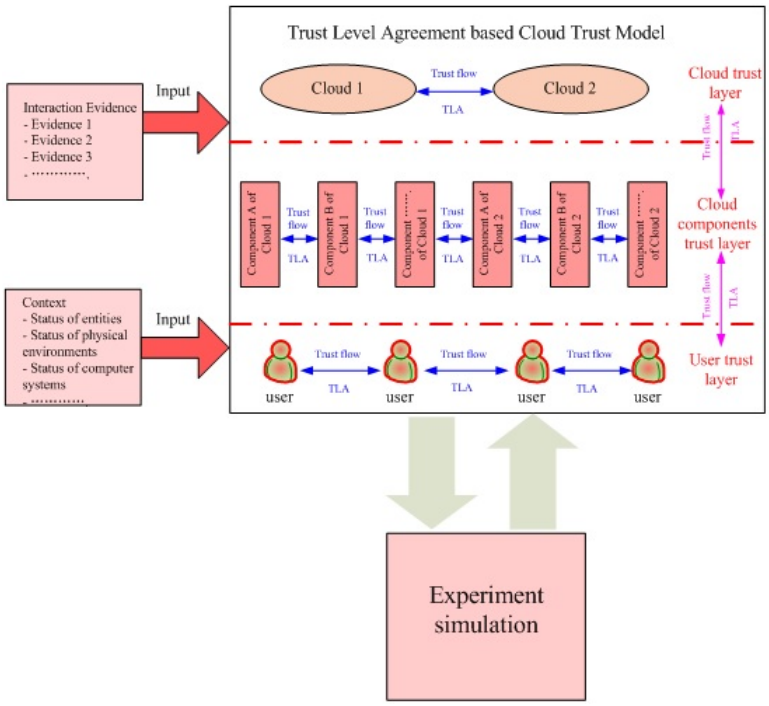


Figure 2: A trust level agreement based cloud trust model

established through the following levels of trust: CSPTL, CCTL and CUTL, so trust evaluation in TLABCTM includes two levels: between the layers and in the same layer. Trust evaluation between the layers involves evaluating the trustworthiness between CSPTL and CCTL and between CCTL and CUTL. Specifically, CSP calculates trust value for other CSP'CC; CC calculates trust value for other CSP; and CC calculates trust value for CU, CU calculates trust value for CC. Trust evaluation in the same layer involves evaluating the trustworthiness in CSPTL and in CCTL and in CUTL. Specifically, CSP calculates trust value for other CSP; CC calculates trust value for other CC; and CU calculates trust value for other CU. We will present the details about how to calculate the trust value in Section 3.3.

In Figure 2, UTLA and PTLA respectively denote User Trust Level Agreement and Provider Trust Level Agreement. UTLA and PTLA are two kinds of TLA agreement. Each CSP and CU respectively maintains UTLA table and PTLA table, which are shown in Tables 1 and 2.

Definition 1. *User Trust Level Agreement (UTLA): It is denoted as* $\{IDEE, UST, UT, ET\}$. *IDE represents the identity of a CU. UST represents requesting service type (operation type and information type,* etc.*). UT represents the trust value of CU. ET represents the lowest expected trust value of requesting service. UTLA shows requesting service type, the lowest expected trust value of requesting service and the trust value of CU, when an entity acts as a CU.*

Definition 2. *Provider Trust Level Agreement (PTLA): It is denoted as* $\{IDE, CCTY, CCPST, CCT\}$. *IDE represents the identity of a CSP. CCTY represents the type of CC. CCPST represents the providing service type of CC. CCT represents the trust value of CC. PTLA shows the providing service type and trust value of CC, when an entity acts as a CSP.*

Table 1: An example of CU_i'$UTLA$ table

IDE	UT	UST	ET
CU_i		Service Type 1	0.3
	0.8	Service Type 2	0.4
		...	...

Table 2: An example of CSP_k'$PTLA$ table

IDE	CCTY	CCT	CCPST
CSP_k	CC_1	0.3	Service Type 1
			...
	CC_2	0.4	Service Type 1
			...
	...	...	...
	CC_j	0.3	...

Consider the situation in Figure 1 where CU_i wants to interact with CSP_k. Based on the proposed TLABCTM, the CU_i will send UTLA to the CSP_k. Then the CSP_k

will estimate whether the CU_i is trusted by checking in UTLA and whether it can provide the service type by checking UST. If the CU_i is trusted and CSP_k'CC_j can provide the service type, CSP_k will send PTLA to the CU_i and select CC_j to provide the service. Then the CU_i will decide whether it should use the service by checking in PTLA. If CCT < ET. the CU_i will refuse to use the service.

3.3 Trust Evaluation of Entity

In this paper, we define trust as an expectation about the behaviors of what an entity, say E_i, expects another entity, say E_j, to perform in a given context. Each entity uses trust values to determine whether it can trust the other entity or not. The trust of entity is represented as a binary value. There are two ways in which to calculate trust value: direct and recommendation. When E_i has enough interaction experience with E_j, E_i uses direct trust to calculate the trust value for E_j. On the other hand, when E_i doesn't have enough interaction experience with E_j, E_i uses recommendation trust to calculate the trust value for E_j. In our paper, an interaction experience threshold is predefines based on the number of interactions in a cloud computing system.

1) Direct trust:
The direct trust value $DT_{E_i}(E_j)$ is defined as:

$$DT_{E_i}(E_j) = \alpha \times \sum_{m=0}^{N(E_j)} \left(\frac{S(E_i,E_j)\times Z}{N(E_j)} + pen(m)\frac{1}{1+e^{-n}}\right) + \beta Risk(E_j). \quad (1)$$

The computing method of direct trust value is proposed by us in the previous work [18]. α and β are weighting factors that satisfies the condition $\alpha+\beta=1$. $N(E_j)$ denotes the total number of interactions that E_i has performed with E_j and $S(E_i,E_j)$ denotes the E_i's satisfaction degree of interaction in its ith interaction with E_j which is in the range of (-1, 1). We use Z to denote the time factor. Thus,

$$Z = \mu(t_m, t_{now}) = \frac{1}{t - now - t_m}, Z \in (0,1) \quad (2)$$

where t_m is the time when the mth interaction occurs and t_{now} is the current time. $pen(m)$ denotes the punishment function and

$$pen(m) = \begin{cases} 1 & \text{if the } m\text{th interaction fails} \\ 0 & \text{if the } m\text{th interaction succeeds} \end{cases}$$

$\frac{1}{1+e^{-n}}$ is the acceleration factor where n denotes the number of failures. It can make trust value drop fast when an interaction fails. As this factor increases with n, it helps avoid heavy penalty simply because of a few unintentional cheats. Finally, $Risk(y)$ is used to express the risk factor.

From Formula (1) we can see that just one time of deception or bad service of trustee may cause its trustor totally distrusts the trustee from then on. If the trustee itself is a just malicious entity, its information or service may not be used and considered by the trustor again; else the trustee does not like to sacrifice the precious and hard established trust value with the trustor.

2) Recommendation trust value
The recommendation trust value $RT_{E_i}(E_j)$ is defined as:

$$RT_{E_i}(E_j) = \sum_{\mu} DT_{E_i}(E_\mu) DT_{E_\mu}(E_j)$$

where $RT_{E_i}(E_j)$ represents the trust that entity E_i places in entity E_j based on asking his friends.

We can write this in matrix notation: If we define DT to be the matrix $[DT_{E_i}(E_\mu)]$ and $\overrightarrow{RT_{E_i}}$ to be vector containing the values RT_{E_i}, then $\overrightarrow{RT_{E_i}} = DT^T\overrightarrow{DT_{E_i}}$. This is a useful way to have each entity gain a view of the cloud computing network that is wider than his own experience. However, the trust values stored by E_i still reflect only the experience of E_i and his acquaintances. In order to get a wider view, E_i may wish to ask his friends' friend $(RT = DT^T)^2 DT_{E_i}$). If he continues in this manner, $(RT = (DT^T)^n DT_{E_i})$, he will have a complete view of the network after n = large iterations. Fortunately, if n is large, the trust vector $\overrightarrow{RT_{E_i}}$ will converge to the same vector for every entity E_i. In other words, $\overrightarrow{RT}$ is a global trust vector in this model. Its elements, RT_{E_μ} quantify how much trust the system as a whole places entity E_μ.

The mechanism of computing recommendation trust allows entities to calculate a recommendation trust for other entities with the recommendation information which is collected by flooding reference trust requests to entities' friends. However, in a large scale cloud computing environment, the mechanism is not scalable due to message overhead problem. From the perspective of sociology, the evidence of trust evaluation between individuals is from direct interaction experience and others' recommendation, but not all others' recommendation information must be collected. According to people's experience of cognitive psychology, old knowledge has less infection and new knowledge has more contribution to trust decision. That is to say, trust value has the attribute of dynamic attenuation over time decay. The trust dynamic nature refers to the trust value of an entity on another entity changes over time due to newer interactions, so the recommendation information should come from newer interactions. The entities in the same layer have the same interaction scenarios and similar interaction requirement,

so the recommendation information from the same layer has a higher accuracy than the one from the other layer.

Based on the above description, we defines an Available Recommendation Entity Set (ARES) to decrease the number of recommendation entity.

Definition 3. *Available Recommendation Entity Set (ARES): The recommendation entity in ARES must satisfy the following three conditions: 1) Recommendation information of the recommendation entity should come from newer interactions; 2) The recommendation entity's trust value shall exceed a trust threshold value T (T = 0.5 in the paper); 3) The recommendation entity is at the same layer with the requesting entity sending reference trust request.*

4 Experimental Study

In this section, in order to evaluate the effectiveness of TLABCTM, a series of test scenarios are developed. The platform of simulation environment is CloudSim toolkit [2] which is a simulation platform based on Java, which supports modeling and simulation of large scale cloud computing data centers. Therefore, it is feasible to simulate our proposed model of cloud computing environments by CloudSim. Each service provider possesses a set of cloud components and the set sizes of all service providers are uniformly distributed (*i.e.* from 1 to 10). One or more components can be combined to produce a cloud service. We generate 100 service providers and 10000 cloud users and 50 kinds of different cloud components. All entities are divided three types: (1) Virtuous entities P_v that provide honest and accurate recommendation data about the other entity; (2) Random entities P_R that provide random recommendation; (3) Malicious entities P_M that provide malicious and false recommendation. We choose to use four metrics, the accuracy rate, response time of trust computing, interaction success rate and change of trust result, to evaluate the performance of TLABCTM, T-Broker [10] and CloudArmor [14]. All simulations were conducted over 1000 sessions. Table 3 shows the parameters used in our experiments. We assume the scenario: Cloud users consume services offered by the service providers. Occasionally, users may require new services, which need other service providers as support. In this case, cloud service provider may contact other service providers to form a collaborative group to share components to fulfill new service requirements. Note that some service providers may reject the collaboration invitation due to various reasons such as limited profits, etc. If no service providers are willing to collaborate, the collaborative group will not be formed.

4.1 Evaluation of Trust Accurate Rate

In the first experiment, we evaluate trust accuracy rate which means the rate of obtaining correct trust results through trust management model on the precondition that all the trust management tasks assigned are completely accomplished within its deadline. The trust accurate rate is compared in TLABCTM, T-Broker [10] and CloudArmor [14]. We submit 100 tasks to 10000 cloud users in order to evaluation certain cloud component. As showed in Figure 3, with the increase of the malicious rate (the percentage of Malicious entities P_M), the TLABCTM can ensure trust accuracy rate in a relatively high level, even when malicious rate is up to 55%, trust accuracy rate is still above 83.5%, and thus it proves the advantage of our model on preventing the behavior of associated cheat of users.

Table 3: Default simulations parameters in the experiments

Service providers	100
Cloud users	10000
Cloud components	50
The rate of virtuous entities	0% - 30%
The rate of random entities	0% - 50%
The rate of malicious entities	0% - 50%

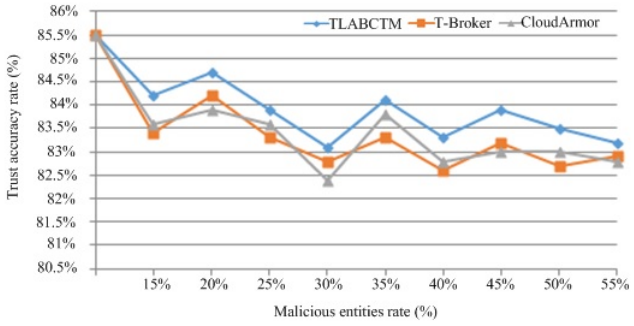
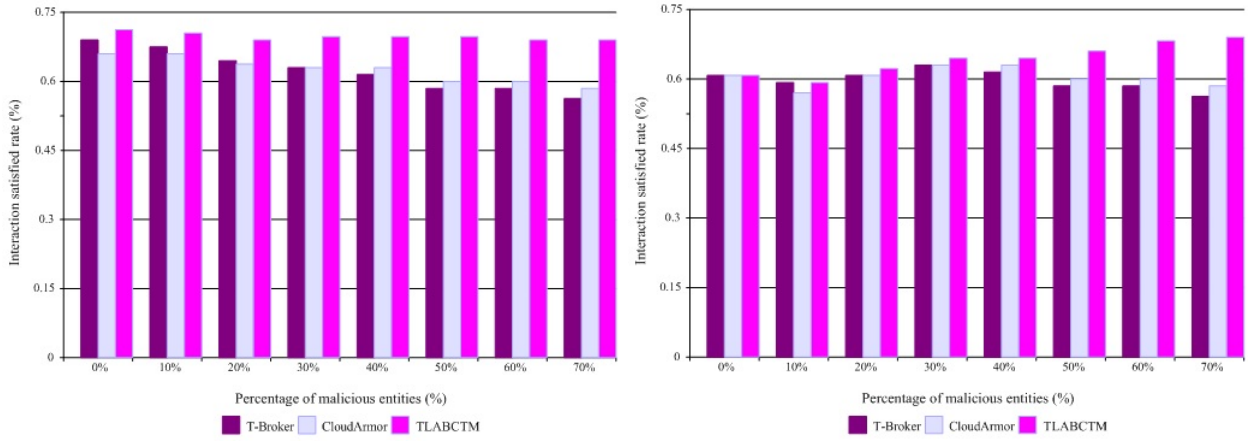


Figure 3: Trust accuracy rate

4.2 Interaction Satisfied Rate

Interaction satisfied rate expresses the satisfied rate in worst case scenario when a cloud user interacts with cloud service providers for getting cloud services. The interaction satisfied rate (ISR): if cloud user has $Number_{total}$ interactions and $Number_{satisfy}$ of them are interactions with satisfied feedback, then $ISR = \frac{Number_{satisfy}}{Number_{total}}$. The interaction satisfied rate is evaluated in the group of experiments. We add a number of malicious servers to the network such that malicious providers make up between 0% and 70% of all servers in the network. For each fraction in steps of 10% we run experiments under two attack models separately and depict the results in in Figure 4a and Figure 4b. We observed a 70% interaction satisfied rate of our mechanism at least in Figure 4a and Figure 4b. For independent cheat and group cheat, our scheme



(a) Independent cheat (b) Group cheat

Figure 4: Simulation results of entities under cheat

performs well even if a majority of malicious providers is present in the network at a prominent place. Even if no malicious providers are present in the system, providers are evaluated as malicious in 3%-5% of all cases - this accounts for mistakes providers make when providing a service, e.g., by providing the wrong meta-data or creating and sharing an unreadable file. As Figure 4(a) and Figure 4(b) shows, comparing with T-Broker [10] and CloudArmor [14], TLABCTM gets more efficient. The main reason is that TLABCTM use trust level agreement to adapt cloud user's service requirement. Before interacting, a cloud user will give out the lowest expected trust value of requesting service. Only when the trust value of component of cloud providers is higher than the lowest expected trust value, the cloud user will decide to use the cloud service provided by cloud providers.

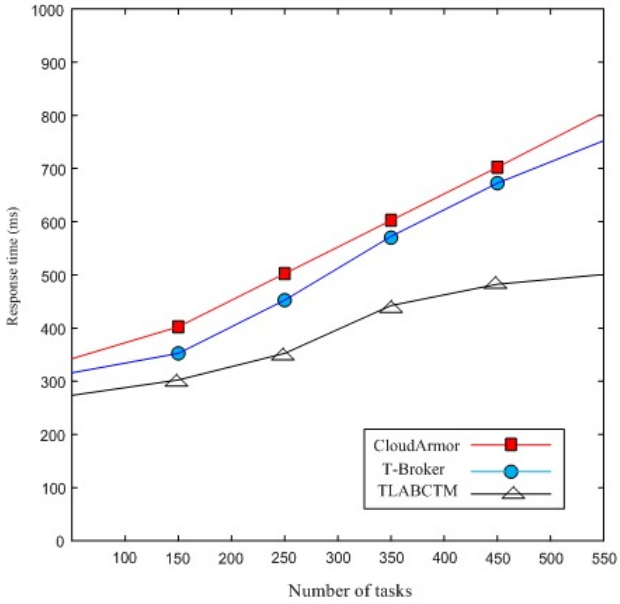


Figure 5: Response time

4.3 Response Time

In the third experiment, we evaluate response time which means the time of obtaining correct trust results through trust management model on the precondition that all the trust management tasks assigned are completely accomplished within its deadline. We submit more than 100 tasks to 10000 cloud users in order to evaluation certain cloud component. The response time is compared in TLABCTM, T-Broker [10] and CloudArmor [14]. From Figure 5 we can see that when T-Broker and CloudArmor are used, the response time is about 750ms and 800ms, while the TLABCTM has the lowest response time 500ms. This is because, by introducing ARES, the service provider is capable of decreasing the number of recommendation entity and eliminating untrustworthy recommendation entity, thus reducing the trust computing time, while there aren't any methods provided to selecting recommendation information in T-Broker and CloudArmor.

5 Conclusions and Future

In the paper we present a cloud trust model based on trust level agreement (TLABCTM), the proposed method not only improves the accuracy of the trust management, and satisfies the trust evaluation requirement of multi-entities in cloud computing environment. In addition, the proposed method can reduce the complexity of trust computing and management and assist cloud computing participants to make good trust decisions. In reality, trust is a social problem, not a purely technical issue. However, we believe that technology can provide the cloud customers with a way of measuring the claims of the cloud service provider as to how trustworthy their clouds are. The benefits of TLABCTM are threefold: First, it presents a hierarchical trust management framework. It divides trust into three layers: cloud trust layer, cloud component trust layer and user trust layer. Trust relationship between

multi-entities is evaluated in TLABCTM. Second, it deals with the dynamic of trust evaluation. TLABCTM classifies the identity of entity, service type, and users can obtain correspond cloud service according to their service requests; and third, it gives a better understanding of cloud components.

Acknowledgements

The work in this paper has been supported by National Natural Science Foundation of China (Program No. 71501156 and No.61373116) and China Postdoctoral Science Foundation (Program No.2014M560796) and special funding for key discipline construction of general institutions of higher learning from Shanxi province.

References

[1] D. S. AbdElminaam, "Improving the security of cloud computing by building new hybrid cryptography algorithms," *International Journal of Electronics and Information Engineering*, vol. 8, no. 1, pp. 40-48, 2018.

[2] R. Buyya, R. Ranjan, and R. N. Calheiros, "Modeling and simulation of scalable cloud computing environments and the CloudSim toolkit: Challenges and opportunities", in *Proceedings of the International Conference on High Performance Computing & Simulation*, pp. 1–11, June 2009.

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

[4] D. C. Edna, O. A. Robson and T. S. J. Rafael, "Trust model for file sharing in cloud computing," in *Proceedings of the Second International Conference on Cloud Computing, GRIDs, and Virtualization*, pp. 66–73, 2011.

[5] D. G. Gopaland R. Saravanan, "Fuzzy based energy aware routing protocol with trustworthiness for MANET," *International Journal of Electronics and Information Engineering*, vol. 3, no. 2, pp. 67–80, 2015.

[6] D. G. Gopaland R. Saravanan, "A novel trust based system to detect the intrusive behavior in MANET," *International Journal of Electronics and Information Engineering*, vol. 3, no. 1, pp. 31–43, 2015.

[7] P. S. Hada, R. Singh and M. M. Meghwal, "Security agents: A mobile agent based trust model for cloud computing," *International Journal of Computer Applications*, pp. 12–15, 2011.

[8] K. Hwang, D. Li, "Trusted cloud computing with secure resources and data coloring," *IEEE Internet Computing*, vol. 14, no. 5, pp. 14–22, 2010.

[9] A. Kumar, K. Gopal and A. Aggarwa, "Design and analysis of lightweight trust mechanism for secret data using lightweight cryptographic primitives in MANETs," *International Journal of Network Security*, vol. 18, no. 1, pp. 1–18, 2016.

[10] X. Y. Li, H. D. Ma, F. Zhou, W. B. Yao, "T-Broker: A trust-aware service brokering scheme for multiple cloud collaborative services," *Information Forensics and Security*, vol. 10, no. 7, pp. 1402–1415, July 2015.

[11] X. Y. Li, L. T. Zhou, Y. Shi, and Y. Guo, "A trusted computing environment model in cloud architecture," in *Proceedings of the Ninth International Conference on Machine Learning and Cybernetics*, pp. 11–14, 2010.

[12] G. Y. Lin, D. R. Wang, Y. Y. Bie, M. Lei, "A mutual trust based access control model in cloud computing," *China Communications*, vol. 11, no. 4, pp. 154–162, 2014.

[13] A. Mosa, H. M. El-Bakry, S. M. Abd El-Razek, S. Q. Hasan, "A proposed E-government framework based on cloud service architecture," *International Journal of Electronics and Information Engineering*, vol. 5, no. 2, pp. 93–104, 2016.

[14] T. H. Noor, Q. Sheng, L. Yao, S. Dustdar, A. H. H. Ngu, "CloudArmor: Supporting reputation-based trust management for cloud services," *IEEE Transactions on Parallel and Distributed Systems*, vol. 27, no. 2, pp. 367–380, 2016.

[15] P. S. Pawar, M. Rajarajan, S. K. Nair, and A. Zisman, "Trust model for optimized cloud services," in *Proceedings of the 6th IFTP International Conference on Trust Management*, pp. 99–112, 2012.

[16] N. Santos, K. Gummadi, and R. Rodrigues, "Towards trusted cloud computing," in *Proceedings of Conference on Hot Topics in Cloud Computing (HotCloud'09)*, June 2009.

[17] N. B. Truong, T. W. Um, G. M. Lee , "A reputation and knowledge based trust service platform for trustworthy social internet of things," in *Proceedings of the 19th International ICIN Conference on Innovations in Clouds, Internet and Networks*, pp. 104–111, 2016.

[18] X. Wu, J. He, F. Xu, "An enhanced trust model based on reputation for P2P networks," in *Proceedings of the IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing (SUTC'08)*, pp. 67–73, 2008.

[19] Z. M. Yang, L. X. Qiao, C. Liu, C. Yang, and G. M. Wan, "A collaborative trust model of firewall through based on cloud computing," in *Proceedings of the 14th International Conference on Computer Supported Cooperative Work in Design*, pp. 329–334, 2010.

[20] W. H. Zhang and H. L. Sheng, "An improved trusted cloud computing platform model based on DAA and privacy CA scheme," in *Proceedings of the IEEE International Conference on Computer Application and System Modeling (ICCASM'10)*, 2010.

Biography

Xu Wu biography. received her Ph.D. degree in Computer Science, from the Beijing University of Technology, in 2010. She was out of post-doctoral stations of the MOEKLINNS Lab, Department of Computer Science and Technology of Xian Jiaotong University in 2016. She is an associate professor of Xi'an University of Posts and Telecommunications. She is working as a visiting scholar in School of Engineering and Technology, Indiana University-Purdue University Indianapolis, Indianapolis, USA, when working on this paper. Her research interests include trusted computing, pervasive computing, mobile computing, and software engineering. She has published more than 50 technical papers and books/chapters in the above areas. Her research is supported by National Natural Science Foundation of China (Program No. 71501156 and No.61373116) and China Postdoctoral Science Foundation (Program No.2014M560796) and Shaanxi Postdoctoral Science Foundation and special funding for key discipline construction of general institutions of higher learning from Shanxi province.

Analysis of One Dynamic Multi-Keyword Ranked Search Scheme over Encrypted Cloud Data

Zhengjun Cao[1], Chong Mao[1], Lihua Liu[2], Wenping Kong[2], Jinbo Wang[3]
(Corresponding author: Zhengjun Cao)
Department of Mathematics, Shanghai University[1]
No.99, Shangda Road, Shanghai, China
(Email: caozhj@shu.edu.cn)
Department of Mathematics, Shanghai Maritime University[2]
Science and Technology on Communication Security Laboratory, China[3].
(Received Mar. 20, 2017; revised and accepted June 26 & July 2, 2017)

Abstract

In 2016, Xia et al. have proposed a scheme for privacy-preserving multi-keyword ranked search over encrypted cloud data [IEEE TPDS, 2016, 340-352]. In this note we show that Xia et al.'s scheme is flawed because the introduced relevance scores do not indicate the true Euclidean distances between the index vectors and the query vector. The scheme has not developed a proper procedure for distance comparison which should be compatible with the technique of Scalar-Product-Preserving Encryption. In the scheme the returned documents are not indeed related to the queried keywords. We also present an improvement using the technique developed by Wong *et al.*'s work [ACM SIGMOD 2009].

Keywords: Cloud Computing; Multi-Keyword Ranked Search; Privacy-Preserving Search; Scalar-Product-Preserving Encryption

1 Introduction

Cloud computing benefits scientific and engineering applications, such as data mining, computational financing, and many other data-intensive activities by supporting a paradigm shift from local to network-centric computing and network-centric content [23]. It enables customers with limited computational resources to outsource large-scale computational tasks to the cloud.

In 2010, Kamara and Lauter [16] discussed the security problem of cloud storage. In 2013, Liu *et al.* [21] explored the problem of multiowner data sharing for dynamic groups in the cloud. Chen *et al.* [12,29] investigated on achieving secure role-based access control on encrypted data in cloud storage. Nabeel *et al.* [24] designed a scheme with privacy preserving policy based content sharing in public clouds.

In 2014, Chen *et al.* proposed two computation outsourcing schemes for linear equations and for linear programming [9, 10]. But the schemes are insecure because the technique of masking a vector with a diagonal matrix is vulnerable to statistical analysis attacks [6]. The Wang *et al.*'s scheme for outsourcing linear equations is flawed [5], too.

In 2016, Khaleel *et al.* [17,25] discussed the possibility of using caching search engine for files retrieval system, and using cloud based technique for blog search optimization. Hsien *et al.* [8, 11, 15, 19] have presented some surveys on public auditing for secure data storage in cloud computing.

Searchable encryption [1–3, 7, 13, 18, 26] is a very appreciated tool that allows a user to securely search over encrypted data through keywords and retrieve documents of interest. Lu *et al.* [22] have discussed how existing additive homomorphic encryption can be potentially used for image search, and proposed two confidentiality-preserving image search schemes based on Paillier's encryption.

In the proposed model, a client has many images and wants to store the image data online for convenient data access anywhere anytime. The client has to encrypt each image and its features and upload the encrypted data to a cloud server. In 2016, Liu and Cao [20] pointed out that Lu *et al.*'s schemes did not make use of the additive homomorphic property at all and the additive homomorphic encryption in one scheme was unnecessary and can be replaced by a more efficient symmetric key encryption.

Recently, Xia *et al.* [28] proposed a scheme for privacy-preserving multi-keyword ranked search over encrypted cloud data. In this note we show that in Xia *et al.*'s scheme the cloud server cannot determine which encrypted index vector I_u is more similar to the encrypted query vector TD. Actually, the relevance score $s_u := I_u \cdot TD = D_u \cdot Q$ does not represent the true similarity between the unencrypted index vector D_u and the unencrypted query vector Q.

The remainder of this paper is organized as follows.

Table 1: Scalar-product-preserving encryption

Key	A $(d+1) \times (d+1)$ invertible matrix M.
DataEnc	For a d-dimensional vector p, set $\hat{p} = (p^T, -0.5\|p\|^2)^T$ and encrypt it as $p' = M^T\hat{p}$.
QueryEnc	For a querying vector q, pick a random number $r > 0$, set $\hat{q} = r(q^T, 1)^T$ and encrypt it as $q' = M^{-1}\hat{q}$.
DistanceComp	Let p'_1 and p'_2 be the encrypted p_1 and p_2 respectively. To determine whether p_1 is nearer to a query q than p_2 is, check whether $(p'_1 - p'_2) \cdot q' > 0$.
DataDecry	Given p', compute $p = (I_d, 0)(M^T)^{-1}p'$ where I_d is the $d \times d$ identity matrix.

In Section 2, we describe the technique of scalar-product-preserving encryption (SPPE) and explain in detail that the technique is compatible with the formal routine of distance-comparison. In Section 3, we provide an explicit description of Xia *et al.*'s scheme (see Table 2). We then point out that Xia *et al.*'s scheme is flawed because the introduced variation of SPPE is not compatible with the routine of distance-comparison (Euclidean distance). In Section 5, we present an improvement of Xia *et al.*'s scheme by extending an index vector to a higher dimension one in order to keep the compatibility between SPPE and distance-comparison. At last, we stress that SPPE must be integrated with the common mechanism for distance comparison in order to represent the similarity scores of vectors.

2 Scalar Product Preserving Encryption

Given two n-dimension vectors X_1, X_2 and another n-dimension vector Y, to determine which $X_i, i = 1, 2$, is more similar to Y, it is usual to compute the distances

$$d(X_i, Y) = \|X_i - Y\| = \sqrt{\|X_i\|^2 - 2X_i \cdot Y + \|Y\|^2},$$

where $i = 1, 2$ and $\|X\|$ represents the Euclidean norm of X, and compare the distances. If $d(X_1, Y) < d(X_2, Y)$, then we assert X_1 is more similar to Y.

In 2009, Wong *et al.* [27] introduced the technique of scalar-product-preserving encryption which can be explained as follows (see Table 1).

The encryption is distance-recoverable because

$$\begin{aligned}
& (p'_1 - p'_2) \cdot q' = (p'_1 - p'_2)^T q' \\
= \; & (M^T\hat{p}_1 - M^T\hat{p}_2)^T M^{-1}\hat{q} \\
= \; & (\hat{p}_1 - \hat{p}_2)^T\hat{q} = \left((p_1^T, -0.5\|p_1\|^2)^T\right. \\
& \left. - (p_2^T, -0.5\|p_2\|^2)^T\right)^T r(q^T, 1)^T \\
= \; & r(p_1^T - p_2^T, -0.5\|p_1\|^2 + 0.5\|p_2\|^2)(q^T, 1)^T \\
= \; & \frac{1}{2}r(2p_1^Tq - 2p_2^Tq - \|p_1\|^2 + \|p_2\|^2) \\
= \; & \frac{1}{2}r\left((\|p_2\|^2 - 2p_2^Tq + \|q\|^2)\right. \\
& \left. - (\|q\|^2 - 2p_1^Tq + \|p_1\|^2)\right) \\
= \; & \frac{1}{2}r\left(\|p_2 - q\|^2 - \|p_1 - q\|\right)^2 \\
= \; & \frac{1}{2}r\left(\|p_2 - q\| + \|p_1 - q\|\right) \\
& \cdot \left(\|p_2 - q\| - \|p_1 - q\|\right) \\
= \; & \frac{1}{2}r\left(d(p_2, q) + d(p_1, q)\right)\left(d(p_2, q) - d(p_1, q)\right)
\end{aligned}$$

Set the similarity score as $s_i = p'_i \cdot q', i = 1, 2$. Since $r\left(d(p_2, q) + d(p_1, q)\right) > 0$, we have

$$(p'_1 - p'_2) \cdot q' > 0 \Leftrightarrow d(p_2, q) - d(p_1, q) > 0,$$

$$s_1 > s_2 \Leftrightarrow d(p_2, q) > d(p_1, q).$$

Thus, the similarity score can be used to indicate the Euclidean distance between the original vector p and the query vector q.

3 Review of Xia *et al.*'s Scheme

The scheme [28] involves three entities: data owner, data user and cloud server.

Data owner has a collection of documents $\mathcal{F} = \{f_1, f_2, \cdots, f_n\}$ that he wants to outsource to the cloud server in encrypted form while still keeping the capability to search on them for effective utilization. Data users are authorized ones to access the documents of data owner. Cloud server stores the encrypted document collection $\mathcal{C}$ and the encrypted searchable tree index $\mathcal{I}$ for data owner.

Upon receiving the trapdoor TD from the data user, the cloud server executes search over the index tree $\mathcal{I}$, and finally returns the corresponding collection of top-k ranked encrypted documents.

The scheme consists of the following phases (see Table 2). We refer to the original for the full description of the scheme [28].

Table 2: Xia *et al.*'s scheme

Date Owner		Server
Setup. Pick a m-bit S and two $m \times m$ invertible matrices M_1, M_2. Set (S, M_1, M_2) as the secret key. Pick a symmetric key encryption $(\mathcal{E}, \mathcal{D})$. **GenIndex**. For files $\mathcal{F} = \{f_1, f_2, \cdots, f_n\}$ and keywords $\mathcal{W} = \{w_1, w_2, \cdots, w_m\}$, set the index $\mathcal{T}$ for $\mathcal{F}$. For the vector D_u in node u, split it into (D'_u, D''_u): if $S[j] = 0$, then $D'_u[j] = D''_u[j] = D_u[j]$; if $S[j] = 1$, then $D'_u[j] + D''_u[j] = D_u[j]$. Set the encrypted index tree as $\mathcal{I}$, where the node u stores $I_u = \{M_1^T D'_u, M_2^T D''_u\}$.	$\xrightarrow{\mathcal{I}, c_i = \mathcal{E}(f_i), i=1,\cdots,n}$	Store $\mathcal{I}$ and all c_i.
Date user		**Server**
Query. Given $\mathcal{W}_q \subset \mathcal{W}$, generate Q for $\mathcal{W}_q$ and split it into Q', Q'': if $S[j] = 1$, then $Q'[j] = Q''[j] = Q[j]$; if $S[j] = 0$, then $Q'[j] + Q''[j] = Q[j]$.	$\xrightarrow{TD = \{M_1^{-1}Q', M_2^{-1}Q''\}}$	**Response**. Compute all scores $s_u = I_u \cdot TD$, return the top ranked id list $\mathcal{C}_{\mathcal{W}_q}$.
Output. Decrypt all files in $\mathcal{C}_{\mathcal{W}_q}$.	$\xleftarrow{\mathcal{C}_{\mathcal{W}_q}}$	

4 Xia *et al.*'s Scheme is Flawed

In Xia *et al.*'s scheme, the cloud server has to compute the relevance score

$$\begin{aligned} s_u &= I_u \cdot TD \\ &= \{M_1^T D'_u, M_2^T D''_u\} \cdot \{M_1^{-1} Q', M_2^{-1} Q''\} \\ &= D'_u \cdot Q' + D''_u \cdot Q'' = D_u \cdot Q \end{aligned}$$

for all nodes. The server then sorts them and returns the top ranked id list $\mathcal{C}_{\mathcal{W}_q}$. We would like to point out that the proposed mechanism fails because the score s_u cannot work well when one considers a true Euclidean distance between the index vector D_u and the query vector Q.

In fact, given two scores $s_i, s_j, i \neq j$, we have

$$s_i - s_j = (D_i - D_j) \cdot Q.$$

If $s_i < s_j$, one cannot determine whether the Euclidean distance $\mathrm{d}(D_i, Q)$ is less than $\mathrm{d}(D_j, Q)$.

Xia *et al.*'s scheme is inspired by Wong *et al.*'s work [27]. The technique of Scalar-Product-Preserving Encryption (SPPE) introduced in [27], *i.e.*, $I_u \cdot TD = D_u \cdot Q$, must be integrated with the routine of Distance-Comparison in order to help the cloud server to sort the final scores according to all $\mathrm{d}(D_u, Q)$. But Xia *et al.* have forgotten to check the compatibility of the variant of SPPE in their scheme with the routine of Distance-Comparison.

5 An Improvement

We now describe an improvement of Xia *et al.*'s scheme by using the technique developed by Wong *et al.* [27]. First, the data owner has to replace S with a $(m+1)$-bit vector rather than the original m-bit vector. Second, the owner sets both M_1, M_2 be of order $m+1$. Third, for the vector D_u in node u, the owner extends it as $\hat{D}_u = (D_u^T, -0.5\|D_u\|^2)^T$. See Table 3 for the details.

The correctness of the improvement is easy to check. In fact, we have

$$\begin{aligned} & s_1 - s_2 = (I_1 - I_2) \cdot TD \\ = \; & \{M_1^T(\hat{D}'_1 - \hat{D}'_2), M_2^T(\hat{D}''_1 - \hat{D}''_2)\} \\ & \cdot \{M_1^{-1}\hat{Q}', M_2^{-1}\hat{Q}''\} \\ = \; & (\hat{D}'_1 - \hat{D}'_2) \cdot \hat{Q}' + (\hat{D}''_1 - \hat{D}''_2) \cdot \hat{Q}'' \\ = \; & (\hat{D}_1 - \hat{D}_2) \cdot \hat{Q} \\ = \; & (D_1^T - D_2^T, -0.5\|D_1\|^2 + 0.5\|D_2\|^2)^T \cdot (Q^T, 1)^T \\ = \; & (D_1 - D_2) \cdot Q - 0.5\|D_1\|^2 + 0.5\|D_2\|^2 \\ = \; & 0.5\left(\|D_2\|^2 - 2D_2 \cdot Q + \|Q\|^2\right) \\ & -0.5\left(\|Q\|^2 - 2D_1 \cdot Q + \|D_1\|^2\right) \\ = \; & 0.5\left(\|D_2 - Q\|^2 - \|D_1 - Q\|^2\right) \\ = \; & 0.5\left(\|D_2 - Q\| + \|D_1 - Q\|\right) \\ & \cdot \left(\|D_2 - Q\| - \|D_1 - Q\|\right) \end{aligned}$$

Table 3: An improvement of Xia *et al.*'s scheme

Date owner		Server
Setup. See the original except that *S is replaced by a* $(m+1)$*-bit vector*, and *both* M_1, M_2 *are of order* $m+1$. **GenIndex**. For the vector D_u in node u, *extend it as* $\hat{D}_u = (D_u^T, -0.5\|D_u\|^2)^T$ split it into $(\hat{D}'_u, \hat{D}''_u)$: if $S[j]=0$, then $\hat{D}'_u[j] = \hat{D}''_u[j] = \hat{D}_u[j]$; if $S[j]=1$, then $\hat{D}'_u[j] + \hat{D}''_u[j] = \hat{D}_u[j]$. Set the tree as $\mathcal{I}$, where the node u stores $I_u = \{M_1^T\hat{D}'_u, M_2^T\hat{D}''_u\}$.	$\xrightarrow{\mathcal{I}, c_i=\mathcal{E}(f_i), i=1,\cdots,n}$	Store $\mathcal{I}$ and all c_i.
Date user		**Server**
Query. Given Q, extend it as $\hat{Q} = (Q^T, 1)^T$ and split it into into $\hat{Q}', \hat{Q}''$: if $S[j]=1$, then $\hat{Q}'[j] = \hat{Q}''[j] = \hat{Q}[j]$; if $S[j]=0$, then $\hat{Q}'[j] + \hat{Q}''[j] = \hat{Q}[j]$.	$\xrightarrow{TD=\{M_1^{-1}\hat{Q}', M_2^{-1}\hat{Q}''\}}$	**Response**. Compute all scores $s_u = I_u \cdot TD$, return the top ranked id list $\mathcal{C}_{\mathcal{W}_q}$.
Output. Decrypt all files in $\mathcal{C}_{\mathcal{W}_q}$.	$\xleftarrow{\mathcal{C}_{\mathcal{W}_q}}$	

Thus,

$$s_1 > s_2 \Leftrightarrow \|D_2 - Q\| > \|D_1 - Q\|.$$

In such case the server can determine that D_1 is nearer to Q than D_2, although D_1, D_2, Q are still unknown to the server.

Xia *et al.*'s scheme [28] is similar to Cao *et al.*'s scheme [4]. Both two schemes are the variations of Wong *et al.*'s scheme [27] except the method to build the unencrypted index vector for each file. But the two schemes failed to develop the technique to integrate the scalar-product-preserving encryption with the routine of distance-comparison (Euclidean distance). The improvement adopts the method developed in [27] and split a vector into two parts. It then encrypts these two parts using two invertible matrixes. The mechanism is useful to resist statistical attacks [14]. This strengthens the security at the expense of a little computational cost.

6 Conclusion

We show that Xia *et al.*'s scheme is flawed and present a possible improvement. We also point out that it is conventional to compare the Euclidean distances between a set of encrypted vectors and a given encrypted vector so as to determine their similarities. We would like to stress that the technique of Scalar-Product-Preserving Encryption must be integrated with the common mechanism for distance comparison in order to represent the similarity scores of these vectors.

Acknowledgements

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

References

[1] M. Abdalla *et al.*, "Searchable encryption revisited: Consistency properties, relation to anonymous ibe, and extensions," *Journal of Cryptology*, vol. 21, no. 3, pp. 350–391, 2008.

[2] D. Boneh *et al.*, "Public key encryption with keyword search," in *Proceedings of International Conference on Theory and Applications of Cryptographic Techniques (EUROCRYPT'04)*, pp. 506–522, May 2004.

[3] D. Boneh *et al.*, "Public key encryption that allows pir queries," in *Advances in Cryptology (CRYPTO'07)*, pp. 50–67, 2007.

[4] N. Cao *et al.*, "Privacy-preserving multi-keyword ranked search over encrypted cloud data," *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 1, pp. 222–233, 2014.

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

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

[7] Y. C. Chang and M. Mitzenmacher, "Privacy preserving keyword searches on remote encrypted data," in *Proceedings of Third International Conference on Applied Cryptography and Network Security (ACNS'05)*, pp. 442–455, June 2005.

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

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

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

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

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

[13] R. Curtmola *et al.*, "Searchable symmetric encryption: Improved definitions and efficient constructions," in *Proceedings of 13th ACM Conf. Computer and Communication Security (CCS'06)*, pp. 79–88, Nov. 2006.

[14] D. Hankerson., A. Menezes, and S. Vanstone, *Guide to Elliptic Curve Cryptography*, Springer, 2004.

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

[16] S. Kamara and K. Lauter, "Cryptographic cloud storage," in *Proceedings of 14th International Conference on Financial Cryptography Data Security (FC'10)*, pp. 136–149, Jan. 2010.

[17] M. Khaleel, H. El-Bakry, and A. Saleh, "A new efficient files retrieval system using caching search engine," *International Journal of Electronics and Information Engineering*, vol. 4, no. 1, pp. 22–31, 2016.

[18] J. Li *et al.*, "Fuzzy keyword search over encrypted data in cloud computing," in *Proceedings of 29th IEEE International Conference on Computer Communications (INFOCOM'10)*, pp. 441–445, Mar. 2010.

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

[20] L. H. Liu and Z. J. Cao, "Analysis of two confidentiality-preserving image search schemes based on additive homomorphic encryption," *International Journal of Electronics and Information Engineering*, vol. 5, no. 1, pp. 1–5, 2016.

[21] X. Liu, Y. Zhang, B. Wang, and J. Yang, "Mona: Secure multiowner data sharing for dynamic groups in the cloud," *IEEE Transactions on Parallel and Distributed Systems*, vol. 24, no. 6, pp. 1182–1191, 2013.

[22] W. J. Lu, A. L. Varna, and M. Wu, "Confidentiality-preserving image search: A comparative study between homomorphic encryption and distance-preserving randomization," *IEEE Access*, no. 2, pp. 125–141, 2014.

[23] D. Marinescu, *Cloud Computing Theory and Practice*, Elsevier, 2013.

[24] M. Nabeel, N. Shang, and E. Bertino, "Privacy preserving policy based content sharing in public clouds," *IEEE Transactions on Knowledge and Data Engineering*, vol. 25, no. 11, pp. 2602–2614, 2013.

[25] J. Singh, "Cloud based technique for blog search optimization," *International Journal of Electronics and Information Engineering*, vol. 4, no. 1, pp. 32–39, 2016.

[26] D. Song, D. Wagner, and A. Perrig, "Practical techniques for searches on encrypted data," in *Proceedings of IEEE Symp. Security and Privacy (IEEE S&P'00)*, pp. 44–55, May 2000.

[27] W.K. Wong, D.W. Cheung, B. Kao, and N. Mamoulis, "Secure knn computation on encrypted databases," in *Proceedings of 35th ACM SIGMOD Int'l Conference on Management of Data*, pp. 139–152, June 2009.

[28] Z. H. Xia, X. H. Wang, X. M. Sun, and Q. Wang, "A secure and dynamic multi-keyword ranked search scheme over encrypted cloud data," *IEEE Transactions on Parallel and Distributed Systems*, vol. 27, no. 2, pp. 340–352, 2016.

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

Biography

Zhengjun Cao is an associate professor with the Department of Mathematics at Shanghai University. He received his Ph.D. degree in applied mathematics from

Academy of Mathematics and Systems Science, Chinese Academy of Sciences. He served as a post-doctor in Computer Sciences Department, Université Libre de Bruxelles, from 2008 to 2010. His research interests include cryptography, discrete logarithms and quantum computation.

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

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

Wenping Kong is currently pursuing her M.S. degree from Department of Mathematics, Shanghai Maritime university. Her research interests include combinatorics and cryptography.

Jinbo Wang received his Ph.D. degree in applied mathematics from Shanghai University. His research interests include applied cryptography and network security.

CP-ABE for Selective Access with Scalable Revocation: A Case Study for Mobile-based Healthfolder

Divyashikha Sethia[1], Huzur Saran[2], and Daya Gupta[1]
(Corresponding author: Divyashikha Sethia)
Department of Computer Science and Engg., Delhi Technological University, India[1]
Shahbad Daulatpur, Main Bawana Road, Delhi, 110042
(Email: sethiadivya@gmail.com)
Department of Computer Science and Engg., Indian Institute of Technology, Delhi[2]
Hauz Khas, New Delhi, Delhi 110016
(Received Mar. 19, 2017; revised and accepted May 16 & June 5, 2017)

Abstract

With the recent advancement in computational and storage capabilities on mobile devices and Internet of Things (IoT), Ciphertext policy Attributed-based Encryption (CP-ABE) can provide confidentiality and direct selective fine-grained access control. There must be an ease of maintaining ciphertext, capability to share and protection against breach of trust. We present a novel revocation scheme Scalable Proxy-based Immediate Revocation for CP-ABE (SPIRC) which does not require prior revocation list, re-encryption and re-distribution of keys. It improves the Proxy-based Immediate Revocation of ATTribute based Encryption (PIRATTE) scheme for scalable revocation with reduced overheads for proxy data and master key generation. The paper also demonstrates the practical implementation of SPIRC for a case study of a portable Mobile-based Healthfolder on a patient mobile device for direct local access as well as sharing with medical professionals using reader application on their mobile devices. The performance evaluation on mid-range Android devices indicates acceptable overheads for access and security.
Keywords: CP-ABE; Mobility; RBAC; Scalable Revocation

1 Introduction

Attribute-Based Encryption (ABE) [15] provides fine-grained access control for sharing ciphertext with a group of users. It comprises of a set of plaintext attributes and an access policy to generate the ciphertext and decryption keys so that each user has a different decryption key. ABE has the advantage that users cannot aggregate their attributes together to decrypt the ciphertext and hence, it is collusion-free. There are several variations of ABE [22, 28] such as Key-Policy Attribute-Based Encryption (KP-ABE) scheme, Ciphertext-Policy Attribute-Based Encryption (CP-ABE) and Hierarchical Attribute-Based Encryption scheme (HABE). The CP-ABE) [8] variation associates a set of attributes to the decryption key and the access policy to the ciphertext. A decryption key can decrypt the ciphertext only if it's associated attributes satisfy the access policy of the ciphertext. Users can be assigned different decryption keys, with each decryption key associated with a subset of attributes that satisfies the ciphertext's access policy. Since CP-ABE can provide Role-based access control (RBAC) by representing a set of attributes for a specific role, we choose it for selective sharing of ciphertext. It also supports revocation as well as collusion resistance. However, it lacks support for scalability as discussed in the survey comparison by Lee *et al.* [22]. CP-ABE has been used for several cloud-based data sharing applications such as for health [23] as well as proved feasible on resource-constrained portable devices such as mobile devices [3] and Internet of Things (IoT) [12].

Motivation: Cloud-based storage solutions are prone to security threats and may not provide 24/7 support in the case of an outage or lack of infrastructure. There is an increase in the penetration of smartphones across the globe. Hence personal portable mobile devices may retain highly available critical data such as that for health [29] and finance and share it directly with other users.

In this paper, we present a case study of a secure portable mobile-based healthfolder on a patient mobile device to store dispersed health data and share it directly with other health professionals. It is a future health management system which can improve availability, sharing capability and mobility to seek the right diagnosis and

treatment across various hospitals.

Health records may be dispersed due to patients visiting various hospitals and hence increase overheads for health management. Developing countries like India lack proper healthcare policies and infrastructure required for a centralised health system. Hence, people visit various hospitals for seeking specialised consultations and second opinions for a reliable diagnosis which leads to dispersed health records.

Health management systems in developed countries are well established. Patients are associated with a particular healthcare or insurance policy such as NPfIT system in U.K [31] and Taiwan Electronic Medical Record Template (TMT) suggested by Chen *et al.* [9]. However, records may be dispersed in the case of citizen mobility for work and tourism across the various states and countries. Also for an emergency situation, a patient may land in a hospital which is not under his health policy. Developed countries have strict and structured health policies which may cause challenges in integrating dispersed health records on cloud-based solutions. Hence, a portable device with health records can benefit patient for high availability as suggested by Anciaux *et al.* [4]. A mobile-based healthfolder can provide mobility to patients to seek efficient treatment and retain their health records securely on their personal devices for both developed and developing nations. Section 5 discusses the case study of the mobile-based healthfolder in detail.

Problem: Since the mobile device is vulnerable to security and privacy threats; it is important to maintain confidentiality and allow selective sharing with authorised users. This paper considers schemes based on Bethencourt *et al.*'s CP-ABE [8] to retain and share secure data on a mobile device since it has been implemented and proved feasible on mobile devices and IoT [3, 12]. The owner of the portable device must access it locally and directly share with other authorised users using selective access policy. There must also be protection from malicious users using a revocation scheme with minimal overheads. We identify the following requirements for retaining and selective sharing data on a portable device using CP-ABE:

R1: No prior knowledge of the revocation list.
There must be no prior requirement for a revocation list for encryption so that the ciphertext can be shared with multiple users.

R2: No re-encryption of ciphertext.
There must be no requirement for re-encryption of ciphertext after revocation so that the owner and other authorised non-revoked users can access it without interruption.

R3: No re-distribution of decryption keys.
There must be no requirement for re-distribution of decryption keys after revocation so that the owner and other non-revoked users can continue to access the ciphertext without interruption.

R4: Revoke a scalable number of users.
The owner must be able to share ciphertext with multiple users as well as revoke a scalable number of malicious users.

R5: Independent of the ciphertext.
No ciphertext specific data must be maintained for user revocation to reduce storage and revocation overheads.

There are several revocation schemes for sharing data on the cloud such as suggested by the survey by Liu *et al.* [24]. However, they do not consider the issues of avoiding re-encryption or key re-distribution after revocation. The revocation schemes can be categorised as direct, indirect and hybrid revocation as discussed by Pang *et al.* [28]. Unlike direct schemes, the indirect schemes do not require any prior knowledge of a revocation list and support broadcast of an intermediate key update, such that only non-revoked users can update their keys. Hence, they are suitable for portable devices to provide ease and flexibility to the owner. They also require a key update phase which can provide bottleneck for interaction with the Certified Authority (CA). Proxy-based Immediate Revocation of Attribute-based encryption (PIRATTE) [21] by Jahid *et al.* is an indirect revocation scheme for CP-ABE which satisfies all of the above requirements except *R4* for scalability. Hence, there is a need to improve PIRATTE for scalable revocation for secure storage and sharing of critical data on a portable device.

Our Contribution:

- Design and implementation of a novel scheme called *Scalable Proxy-based Immediate Revocation for CP-ABE (SPIRC)* which extends PIRATTE [21] for scalable user revocation. It fulfils all revocation requirements *R1-R5* for sharing of secure data from portable devices. The scheme requires a trusted proxy-based server which manages cryptographic credentials for registered owner and users and also provides proxy data to users to complete decryption. The proxy server updates the proxy data for a revoked user so that decryption fails. Section 3.1 describes the details of the trusted server in the intuition.
- Demonstration of a case study for a next-generation Portable Mobile-based Healthfolder [29] on a patient mobile device which retains dispersed health records from various hospitals. The patient can share it directly with health professionals based on their roles such as a doctor, nurse, lab technician and a pharmacist. The Healthfolder is encrypted using the SPIRC scheme for selective access by health professionals and scalable revocation. The health professionals access it directly with their mobile devices as shown in Figure 1.
- Practical implementation and evaluation of SPIRC for the prototype of the Mobile-based Healthfolder

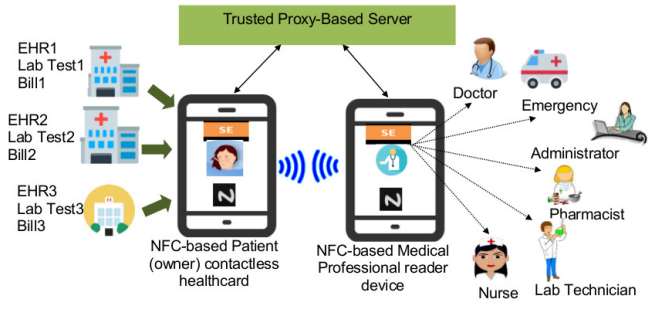


Figure 1: Mobile-based Healthfolder stakeholders: Patient, External users and trusted proxy-based server

on mid-range Android devices. Performance evaluation indicates acceptable delays for communication and security handshake. A comparison of different schemes for storage and computational overheads shows that SPIRC provides scalable revocation with lower overheads for proxy data and master key generation.

To the best of our knowledge, our work is the first novel attempt to address the issues of selective sharing and scalable revocation for a portable device using Bethencourt *et al.*'s CP-ABE scheme [8]. In future, we can work on scalable revocation schemes based on variations of CP-ABE with better performance such as Cheung *et al.*'s provably secure CP-ABE [10] and Lewko *et al.*'s CP-ABE scheme based on (Linear Secret Sharing Scheme) LSSS matrix [7].

The rest of the paper comprises of Related Work in Section 2, details of the new SPIRC scheme in Section 3 and Security Analysis of SPIRC in Section 4. Section 5 presents a Case Study for Selective Access for Portable Mobile-based Healthfolder along with its Security Analysis with SPIRC and Implementation and Performance Evaluation. It is followed by performance comparison of revocation schemes in Section 6. The paper finally concludes with Section 7 for Conclusion and Future work.

2 Related Work

The indirect revocation schemes for CP-ABE for portable devices must satisfy all revocation requirements *R1-R5* for the ease of portability, personal access for the owner and sharing data directly with other external authorised users.

CP-ABE techniques used in the cloud-based record sharing schemes such as those for health records are not directly suitable for portable devices. Narayan *et al.* [26] propose a broadcast variation of CP-ABE which has the limitation that the length of ciphertext grows proportionally with the number of revoked users. Hence, this may not be feasible for portable devices with limited storage. Liet *et al.* [23] suggest a scalable Electronic Health Record (EHR) scheme which uses revocation scheme by Worcester *et al.* [33] which requires re-encryption for revocation and violates requirement *R2* for a portable device.

Attrapadung and Imai [5] provide a hybrid revocation scheme which supports both direct and indirect modes. However, it has the drawback of longer user secret key length, which can be difficult to store on a portable mobile device. Ibraimi *et al.* [19] suggest an indirect revocation scheme which generates two portions of the private key one of which is retained by the user and the other with a mediator. The mediator sends the right portion of the key to a user only if it not revoked. However, it uses CP-ABE scheme by Cheung *et al.* [10] which has the drawback that there is an increase in the size of ciphertext and key with the increase in the total number of attributes in the access policy. Hence it is not suitable for a mobile device with limited storage. Modi *et al.* [25] propose a revocation scheme for secure file access on the cloud. However, it violates requirement *R1* needed for scalable sharing of ciphertext on a portable device. Hur *et al.* [17] propose an indirect revocation scheme to provide fine-grained attribute revocation with the limitation of requiring re-encryption of ciphertext and hence violates the requirement *R2*.

PIRATTE (Proxy-based Immediate Revocation of ATTribute based Encryption) [21] scheme by Jahid *et al.* is a variation of Bethencourt *et al.*'s CP-ABE [8], which provides indirect revocation without re-encryption of the ciphertext and key re-distribution. Users receive proxy data from the proxy-based server to complete decryption. PIRATTE scheme uses a polynomial P of degree $t+1$ in the master key. The trusted server divides the secret $P(0)$ into portions and provides a share to each user. During decryption, each user seeks a proxy key and t shares of the secret from the proxy-based server. It uses Lagrange's interpolation to combine the t secret portions with the user portion to generate the secret $P(0)$. If the user is non-revoked, the proxy-based server sends valid secret portions. Otherwise, it sends invalid secret portions, so that the user cannot generate the secret $P(0)$ and hence decryption fails. PIRATTE fulfils all revocation requirements, except for *R4* since it can revoke only limited t number of users.

A permanent revocation scheme (referred to as PERMREV in this paper) by Dolev *et al.* [13], modifies the Bethencourt *et al.*'s CP-ABE [8] scheme and associates a counter CTR with the ciphertext and a user state $State_i$ for the ith user $user_i$. It considers ciphertext to reside on a secure cloud-based system. For revocation of $user_i$, the secure server updates CTR, re-encrypts the ciphertext and sends the updated $State_i$ with new CTR only to the non-revoked users. Since revoked users do not receive any updated state, decryption fails. To avoid ciphertext re-encryption a Modified PERMREV scheme referred as M-PERMREV in this paper requires a server to broadcast $State$ to all users. For a revoked user, the server updates the CTR and the user state for only revoked users, which causes failure of decryption. However, M-PERMREV scheme does not fulfill requirement *R5* since it associates a constant CTR with the user's

state $State_i$ for every ciphertext.

CP-ABE can provide RBAC as suggested by role-based access control scheme (RACS) for sharing medical data on cloud by Tian *et al.* [32]. However it cannot be used for portable devices since it does not fulfill requirement *R2*.

The SPIRC scheme presented in this paper improves PIRATTE for scalable revocation and fulfils all requirements *R1-R5* for revocation.

3 Scalable Proxy-based Immediate Revocation For CP-ABE Scheme

Billinear pairings. Let G_1, G_2 and G_T be multiplicative cyclic groups of prime order p. Let g_1 and g_2 be a generator of G_1 and G_2 respectively. e is a bilinear map such that $e : G_1 \times G_2 \rightarrow G_T$. It has the following properties:

1) **Bilinearity:** for all u, v element of G_1, G_2 and a; b element of Z_p, $e(u^a, v^b) = e(u, v)^{ab}$.

2) **Non-degeneracy:** $e(g, g) \neq 1$.

Intuition: This paper looks into the issue of storing secure data on a portable device and sharing it using selective access control. It outsources encryption to a trusted proxy-based server. The owner decrypts the ciphertext locally on the personal portable device, and shares it directly with other users who decrypt it locally on their respective devices.

The trusted proxy-based server retains credentials and identities of registered users, as well as constants related to the proxy data.

Each $user_i$ registers with a trusted proxy-based server and is associated with a set of random parameters $S_i = \{\lambda_i, a_i, b_i\}$. The constants are associated with the decryption keys of the user as well as proxy data. For decryption, a user contacts the trusted server through a secure channel such as HTTPS to gather proxy data to complete the decryption process. The trusted server also maintains a revocation list RL which is populated by an authorized owner or an administrative personnel to protect portable device from malicious users on breach of trust or theft of device. To revoke a user, the proxy-based server updates S_i so that the proxy-based data is modified and causes decryption to fail.

The cloud-based service is contacted only for seeking proxy data and not for the actual ciphertext as in the cloud-based sharing applications [23]. The trusted proxy-server must comply to all requirements for Trusted Computing [30]. The trust between authorized users and proxy-server can be established through some of the existing techniques such as mutual authentication and remote attestation techniques as suggested in [6], to ensure that they are not compromised with any malicious software. Further authorized users can communicate with the trusted server using separate CP-ABE access policies for RBAC for allowing trusted revocation and configurations of credentials by users with administrative roles. This can ensure secure maintenance of credentials as well as revocation list on the proxy-server. The detailed design of the trusted proxy-based server are beyond the scope of the paper.

3.1 SPIRC Construction

The SPIRC scheme supports scalable user revocation without requiring re-encryption or re-distribution of keys. This paper modifies Jahid *et al.*'s PIRATTE [21] scheme for scalable revocation for infinite users. It comprises of the following algorithms:

Setup: Generates Public key PK and Master key MK.

Encrypt(PK, M, τ): Takes data M, Public key PK, and access policy τ to generate the ciphertext CT.

KeyGen(MK, S): Takes master key MK and set of attributes S and generates the secret key SK.

Proxy-Data(U_k, RL): Takes user identity u_k and the revocation list RL as input and generates the proxy data PXD. It also invokes CONVERT function to transform portion of the ciphertext C'_x for each attribute x satisfied by users u_k and generates the converted portion C''_x.

Decrypt(CT, SK): Decrypts the ciphertext CT to generate a plaintext M if the set of attributes S in SK satisfy the access policy τ that is used to generate ciphertext CT.

The details of the different phases are given below:

Setup. The trusted proxy-based server chooses G_1, G_2, g_1, g_2 and random elements α and $\beta \in Z_p$ to generate the public key PK and a master key MK.

$$\begin{aligned} PK &= G_1, G_1, g_1, g_2, h = g_1^{\beta}, e(g_1, g_2)^{\alpha} \\ MK &= \beta, g_2^{\alpha} \end{aligned}$$

Unlike PIRATTE, for master key MK, there is no generation of polynomial P of degree $t + 1$, where t is the number of users that can be revoked. Hence, it provides scalable revocation.

Encrypt(PK, M, τ): The tree structure τ represents the access policy with attributes at leaves and threshold of k-of-n gates at the interior nodes. q_x is the polynomial at node x with degree $d = k$ *-1*, where k is the threshold value of the node. For all OR nodes and leaf nodes, the polynomial degree is 0. The proxy-based server chooses a random secret $s \in Z_p$ for a message M, such that for root node $R, q_R(0) = s$. The secret is distributed from top to bottom for all other nodes, $q_x(0) = q_{parent(x)}(index(x))$, where $index(x)$ is a number associated with x between *1* and *num* (number of children of *parent(x)*). X is the

set of leaf nodes in the access tree τ. The ciphertext CT is: $CT = (\tau, \tilde{C} = Me(g_1, g_2)^{\alpha s}, C = h^s)$, $\forall x \in X : C_x = g_1^{q_x(0)}$, $C'_x = H(att(x))^{q_x(0)} = g_2^{h_x q_x(0)}$. $H : \{0,1\}* \rightarrow G_2$ is a hash function that maps a string attribute to a random element in G_2 and $h_x = \log_{g_2} H(att(x))$.

KeyGen(MK, S): It generates the secret key SK for $user_i$ for a set of attributes S. For each user i, it chooses a random number r along with set $S_i = (\lambda_i, a_i, b_i) \in Z_p$ and for each attribute j it chooses a random number $r_j \in Z_p$. $SK = (D = g_2^{(\alpha+r)/\beta}, \forall j \in S : D_j = g_2^r H(j)^{r_j(\lambda_i a_i + b_i)} = g_2^{r+h_j r_j(\lambda_i a_i + b_i)}, D'_j = g_1^{r_j}, D''_j = (D'_j)^{a_i} = g_1^{r_j a_i})$.

The portions of the secret key SK, D_j and D'_j for each attribute j contain random number r_j and D contains random number r which is specific to a user. Hence attributes from different users cannot be combined together and prevents collusion.

Proxy-Data($user_i$): Proxy-based server maintains a random set S_i for each user along with a revocation list. For the completion of decryption, $user_i$ seeks proxy data PXD from the proxy-based server which is unique for a user.

$PXD = \lambda_i$. The trusted server sends proxy data PXD to $user_i$, who also sends C'_x to the proxy-based server to return Convert C''_x as: $CONVERT(C''_x, b_i) = (C'_x)^{b_i} = g^{h_x q_x(0) b_i}$. The user secret SK is blinded by $(\lambda_i a_i + b_i)$ and needs C''_x along with C_x and C'_x. Proxy can revoke the user by updating the λ_i and b_i for $user_i$ in PXD and C''_x.

Decrypt. For a $user_i$, each leaf node x of the policy is an attribute, with $j = attr(x)$, if $j \in S$, (S is the set of attributes) then, $DecrytpNode = A_j$ is as follows:

$$
\begin{aligned}
A_j &= \frac{e(C_x, D_j)}{e(D''_j, C'_x)^{\lambda_i} e(D'_j, C''_x)} \\
e(C_x, D_j) &= e(g_1^{q_x(0)}, g_2^{r+h_j r_j(\lambda_i a_i + b_i)}) \\
&= e(g_1, g_2)^{q_x(0) r + q_x(0) h_j r_j(\lambda_i a_i + b_i)} \\
A_j &= \frac{e(g_1, g_2)^{q_x(0) r + q_x(0) h_j r_j(\lambda_i a_i + b_i)}}{e(g_1^{r_j a_j}, g_2^{h_j q_x(0)})^{\lambda_i} e(g_1^{r_j}, g_2^{h_j q_x(0) b_i})} \\
&= \frac{e(g_1, g_2)^{q_x(0) r + q_x(0) h_j r_j(\lambda_i a_i + b_i)}}{e(g_1, g_2)^{r_j a_i h_j q_x(0) \lambda_i} e(g_1, g_2)^{r_j h_j q_x(0) b_i}} \\
&= \frac{e(g_1, g_2)^{q_x(0) r + q_x(0) h_j r_j(\lambda_i a_i + b_i)}}{e(g_1, g_2)^{r_j a_i h_j q_x(0) \lambda_i + r_j h_j q_x(0) b_i}} \\
&= \frac{e(g_1, g_2)^{q_x(0) r + q_x(0) h_j r_j(\lambda_i a_i + b_i)}}{e(g_1, g_2)^{r_j h_j q_x(0)(\lambda_i a_i + b_i)}} \\
&= e(g_1, g_2)^{q_x(0) r}
\end{aligned}
$$

Each $user_i$ has associated constant values λ_i, a_i and b_i which are maintained on the proxy-based server. Whenever revocation is required, the proxy-based server updates λ_i or b_i, which are part of PXD and C''_x, and cause the DecryptNode function to fail and return $\perp$.

The rest of the decryption process is the same as in the Bethencourt *et al.*'s CP-ABE scheme [8] to obtain the original message M.

For each node z of a non-leaf node x, it calculates $F_z = e(g_1, e_2)^{r q_q(0)}$. If S_x is the set of children of x so that $F_z \neq \perp$. This is followed by the following decryption process:

$$
\begin{aligned}
F_x &= \prod_{i=1}^{S_x} F_z^{\lambda_i}, (i = index(z) \lambda_i calculated \forall z \in S_x) \\
&= \prod_{i=1}^{S_x} (e(g_1, g_2)^{r q_z(0)})^{\lambda_i} \\
&= \prod_{i=1}^{S_x} (e(g_1, g_2)^{r q_{parent(z)} index(z)})^{\lambda_i} \\
&= \prod_{i=1}^{S_x} (e(g_1, g_2)^{r q_x(i)})^{\lambda_i} \\
&= e(g_1, g_2)^{\sum_{i=1}^{S_x} r q_x(i) \lambda_i} \\
&= e(g_1, g_2)^{r q_x(0)}
\end{aligned}
$$

Let $A = e(g_1, g_2)^{r q_R(0)} = e(g_1, g_2)^{r q_R(0)} = e(g_1, g_2)^{rs}$ at root node R. Decryption can be done as follows, $\frac{\tilde{C}}{\frac{e(C,D)}{A}} = Me(g_1, g_2)^{\alpha s} \frac{e(g_1, g_2)^{rs}}{e(g_1, g_2)^{\alpha s + rs}} = M$.

4 Security Analysis For SPIRC

The definitions for user-based revocation are as per [21].

4.1 Security Game

In the security game between an adversary and a challenger, the encryption remains secure even when the adversary compromises the proxy and obtains it's key after a recent revocation.

Setup. A challenger runs the SETUP and provides public parameters PK to the adversary. Challenger also generates a proxy data PXD.

Phase 1. The adversary performs repeated queries for KEYGEN to obtain keys for multiple user $u_1, \cdots, u_{q_1}$ with different sets of attributes $S_1, \cdots, S_{q_1}$. The adversary also contacts the proxy for the $CONVERT(\{C'_1, \cdots, C'_r\}, u_k)$ for $C'_i \in G_1$. Simultaneously challenger also computes CONVERT with the stored values. The adversary contacts proxy server to get the proxy data PXD. In the meanwhile challenger updates the proxy data PXD.

Challenge. The adversary submits messages M_0 and M_1 of equal lengths and an access structure A* such that either u_k is to be revoked or S_k does not satisfy A*.

The challenger flips a coin to obtain a random bit b and returns M_b encrypted with the access policy A*. It also runs Proxy-Data and returns the proxy data PXD to the adversary.

Phase 2. The adversary makes repeated queries to the KEYGEN to obtain keys for users $u_{q_1+1}, \cdots, u_{q_2}$ with attributes $S_{q_1+1}, \cdots, S_{q_2}$. The new keys are such that if $u_k \notin$ revocation list RL, then S_k does not satisfy A*.

Guess. The adversary outputs a guess b' of b.

The adversary has an advantage defined as $Pr[b' = b] - \frac{1}{2}$. As in PIRATTE even if an adversary $user_j$ finds Proxy portions of another $user_i$, the portions will not help him with the decryption since each user has a different set of random constants values. The SPIRC scheme provides forward secrecy since a revoked user cannot decrypt any previously recorded ciphertext.

4.2 Security Proof

Asymmetric Groups Similar to PIRATTE [21] for user i and attribute j, different groups are used for C'_j and D'_j. The user sends C'_j to convert and receive C''_j, where $C''_j = C_j^{'b_i}$. If both C'_j and D'_j belong to the same group and user sends D'_j to convert, then user will get $D_j^{'b_j} = g_2^{r_j b_j}$. User will also get λ_j and can get $D_j^{''\lambda_j} = g_2^{r_j \lambda_j a_j}$. Combining these two terms by multiplication will provide $g_2^{r_j(\lambda_j a_j + b_j)}$. User can use this to decrypt any ciphertext without using the proxy-based server for revocation. Hence asymmetric pairing is used with different groups for C_j and D'_j.

Similar to PIRATTE [21], SPIRC is based on the generic asymmetric bilinear group model, which considers a asymmetric pairing of $e : G_1 \times G_2 \rightarrow G_T$, with the assumption that their is no isomorphism from G_1 to G_2. Both are based on CP-ABE [8] scheme, and hence are secure against Chosen Plaintext Attack (CPA). Other variations of CP-ABE such as Cheung *et al.*'s CP-ABE [10] are secure against Chosen Ciphertext Attack (CCA). However, this paper focuses on only Bethencourt et.al's CP-ABE scheme which has been proven feasible on mobile devices and IoT devices [3, 12].

Theorem 1. *The construction of SPIRC scheme is secure under the generic bilinear group model. It assumes that there is unexpected collisions between asymmetric groups.*

The paper assumes that in the security game, $A*$ contains single attribute A_j for some attribute j. After Phase 2, the adversary has the following elements for each user u_k and A_j from S_k: $G_1 : g_1, g_1^{\beta}$, $C = g_1^{\beta s}$, $C_j = g_1^s$, $D'_j = g_1^{r_{ukj}}$, $D''_j = g_1^{r_{ukj} a_k}$.

Secret s encrypts the message and $H(j) = g_2^{h_j}$. $G2 : g_2, D = g_2^{(\alpha+r)/\beta}, D_j = g_2^{r_u + h_j r_{ukj}(\lambda_k a_k + b_k)}, C'_j = g_2^{h_j s}$.

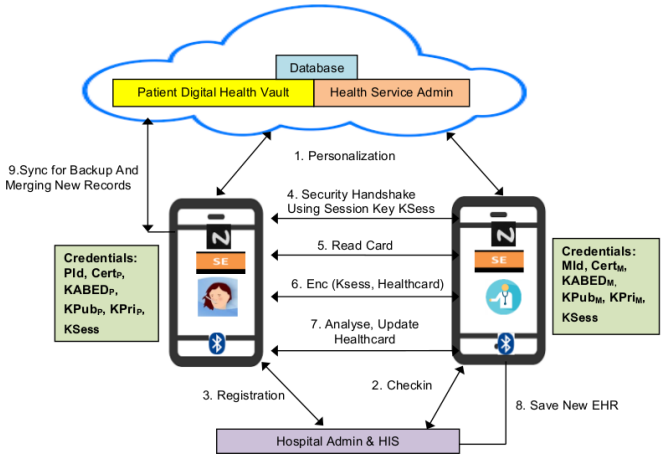


Figure 2: Selective access of a mobile-based healthfolder

$G_T : e(g_1, g_2)^{\alpha}$, $M.\ e(g_1, g_2)^{\alpha s}$.

Adversary only knows u_k for all revoked users in the revocation list $RL*$. However, secret s occurs only in elements of the ciphertext C, C_j and C'_j. To guess s, the adversary can compute $e(C, D^{(uk)}) = e(g_1, g_2)^{\alpha s + r_{uk} s}$. To determine $e(g_1, g_2)^{\alpha s}$, adversary must compute $e(g_1, g_2)^{rs}$. However, it is not feasible to compute it from D_j. Hence it is difficult for the adversary to determine the secret s in the security game. SPIRC is hence secure under the generic asymmetric bilinear group model.

5 Case Study: Selective Access Mobile-based Healthfolder

5.1 System Design

We present the system design for a Mobile-based Healthfolder on a patient device. The SPIRC scheme encrypts it and stores it on a secure storage with direct selective access. Figure 2 shows the system design. Our preliminary work in [29] is based on PIRATTE [21]. SPIRC scheme improves it for scalable revocation for enhanced portability and mobility of a patient across hospitals. The system comprises of a patient's mobile device with a Healthfolder containing different health data from dispersed hospitals.

The Mobile-based Healthfolder is retained as a large sized contactless card using NFC-based Host Card Emulation(HCE) [2]. A health professional accesses the software-based HCE contactless card by a tap of his mobile device, using IoT-based communication interfaces of NFC [11] and Bluetooth. It is supported by a cloud-based HealthSecure service which comprises of a trusted proxy-based server and a secure digital vault to store data sync. The proxy-based server maintains cryptographic credentials, unique user identities and support for SPIRC proxy decryption. The service can be managed by government intuitions, insurance companies or chain of well-known

Table 1: Main terms for mobile-based healthfolder

Term	Description
Idp	Identifier for Patient
Idm	Identifier for Health Professional
H/H'	Unencrypted/Encrypted Healthfolder
U	User (P-Patient/M-Health professional)
CU	**User's Credentials on SE**
{KUpub/KUpri	User's Public/Private RSA keys
Certu	User certificate for {Idu,KUpub}
KDRUabe	User CP-ABE Read decryption key
KDWUabe }	User CP-ABE Write decryption key
$Section_i$	Healthcard ith section, i = 1-7
ri	Random number for $Section_i$
rei	Encrypted ri for ith section E(KEWabe,ri)
RW={re1..re6}	Write policy encrypted random nos.
$Update_i$	Update for $Section_i$
Ksym	Symmetric Session key

Table 2: Healthfolder organization

Sections	1	2	3	4	5	6	7
Stakeholder	Basic Vitals	Allergies /Disease	Advan. Vitals	Medic	Lab/ Immun	Emerg / Admin	Non-clinical
Doctor	RW	RW	RW	RW	RW	R	R
Nurse	RW	R	R	R	R	R	R
Pharmacist	--	--	--	RW	--	R	--
Lab Tests	--	--	--	--	RW	R	--
Emergency	RW	RW	RW	RW	RW	RW	RW
Patient	R	R	R	R	R	R	RW
Admin	R	R	R	R	R	RW	R
Read Policy	ACRB	ACRB	ACRB	ACRM	ACRL	ACRALL	ACRB
Write Policy	ACWBV	ACWSP	ACWSP	ACWM	ACWL	ACWADM	ACWNC

Table 3: Healthfolder CP-ABE write access policies

Name	Section	Policy
ACWBV	Basic vitals	AND(wobb,OR(doctor,nurse,emerg))
ACWSP	Special	AND(wobs,OR(AND(doctor,department),emerg))
ACWM	Medication	AND(wmed,OR(doctor,pharm,emerg))
ACWL	Lab tests	AND(wlab,OR(doctor,pharm,emerg))
ACWADM	Admin	AND(wadm,OR(admin,patient))
ACWNC	Non-clinical	AND(wnoncl,OR(emerg,patient))

hospitals and must have a policy that complies with the requirements for Trusted computing [30].

Both devices of the patient and health professional register with the HealthSecure service and store secure credentials and identity on tamper resistant Secure Element (SE) in the form of a microSD card. It can be accessed internally through applications compiled with special libraries on the processor. The SE utilises Java Card [27] technology which enables Java-based applets to execute with limited memory and processing capabilities.

After an NFC tap between the mobile devices, they mutually authenticate and establish an asymmetric session key *Ksym*. The patient mobile device automates Bluetooth setup over HCE for higher throughput. All subsequent communication is encrypted using *Ksym*. A health professional reads and writes to set of sections on the Mobile-based Healthfolder over Bluetooth and terminates it after the transfer is complete.

Due to the high computational costs of bilinear pairing, the Mobile-based Healthfolder outsources CP-ABE encryption to the HealthSecure service. However, both the Patient and Medic mobile device locally decrypt to view the Healthfolder. The health professional evaluates the past records and provides diagnosis and treatment for the current medical condition. All new updates are written securely to the Mobile-based Healthfolder. Hence a patient retains upto date health records. Table 1 describes the main notations for the case study.

5.2 Selective RBAC with SPIRC

The Mobile-based Healthfolder retains different health records such as prescriptions, reports, medication details from various hospitals in standard formats such as HL7 [16] for interoperability. It organises each department record into different subsections. Various authorised health professionals access them as per their roles with selective RBAC as shown in Table 2.

For each section, a read access policy encrypts it, and a write access policy encrypts a section specific random number *ri* as *rei*. Table 2 shows the different access policies for each section on the Healthfolder. A stakeholder stores two decryption keys: a read decryption key *KDRUabe* and a write decryption key *KDWUabe* to access the authorised sections. A CP-ABE decryption key can decrypt all sections for which the attributes in the key can satisfy the section access policy.

A stakeholder first reads the Healthfolder and obtains the concerned sections by decrypting with his read decryption key *KDRUabe*. However, once he can read a section, he must be able to update it only if he has access according to the write access policy.

Figure 3 shows a sample read access policy *ACRALL* which permits all stakeholders to read. Each section has a different write access policy with a special set of associated attributes as shown in Table 3. For example, to read sections encrypted with *ACRM* and *ACRALL* read policies, a pharmacist must have a read decryption key with attributes that satisfy the related access policies. Similarly, to write to sections encrypted with *ACWM* write policy, the write decryption key must have attributes which satisfy the access policy. For example for a user pharmacist, the decryption key must have attributes *pharmacy*, time between and 4 and *wmed* to satisfy the access policy.

Write Access Policy. For each section *i* of the healthfolder, random number *ri* is encrypted with the write CP-ABE policy of the section as *rei*. When a stakeholder requests to write to section *i*, patient challenges it with the encrypted *rei* for the section. If the

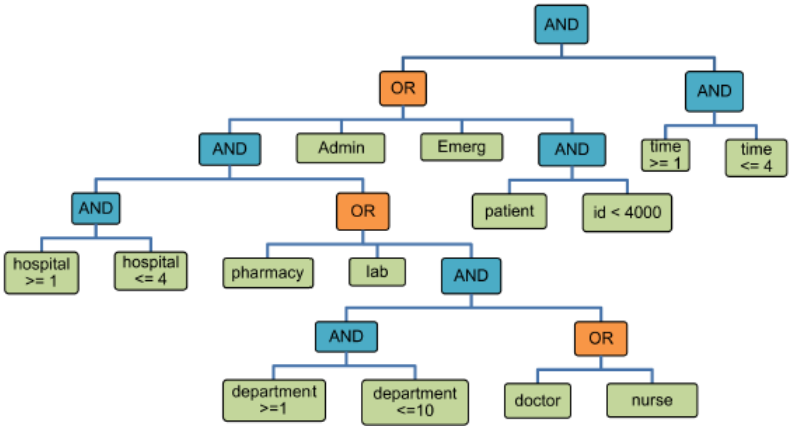


Figure 3: CP-ABE read access policy ACRALL

Table 4: Sequence for SPIRC-based selective access and scalable revocation

S.No	Messages	
1'.	Card:	Personalisation: ((KPpub, KPpri, KDRPabe, KDWPabe, RW=(re1..re7))
1".	Reader:	Personalisation: ((KMpub,KMpri) Non-emergency:KDRMabe, KDWMabe)
2.	Card ⟷ Reader:	Mutual Authentication to generate Ksym
3.	Card ← Reader:	Action: write/read, $Section_i$
4.	Card:	MP1=H' || rei
5.	Card → Reader:	E(Ksym,MP1)
6.	Reader ⟷ Server:	If Emergency personnel obtain BTG keys (KDRMabe, KDWMabe)
7.	Reader ⟷ Server:	Proxy-based server-based decryption H=D(KDRMabe,H'), ri=D(KDWMabe, rei)
8.	Server:	Revoke users in RL
9.	Server:	ri' = ri+1, $Access = hash(Update_i)$, $MM1 = ri'\|\|Update_i$
10.	Card ← Reader:	E(Ksym,MM1)
11.	Server:	If ri'== ri+1 then accept $Update_i$
12.	Card →: Server	$Update_i$ through HTTPS
13.	Server:	Revoke key if user is an Emergency personnel Sync $Update_i$ on digital vault, Re-encrypt H as H"
14.	Card ← Server:	H" through HTTPS

stakeholder has access to write, he can decrypt *rei* using his write decryption key *KDWUabe*. In response, he computes *ri'=ri+1* and sends it to the Mobile-based Healthfolder along with the update $Update_i$ for the section. The Mobile-based Healthfolder compares the received *ri'* and the locally computed value of $(ri+1)$. If they match, then the $Update_i$ is written on the healthfolder, else it is rejected.

Revocation. Healthsecure service associates time-based attributes with the decryption key and each stakeholder must renew it periodically. The *ACRALL* policy in Figure 3 shows the time-based attributes. Decryption keys with time attributes between *1* and *4* will only satisfy this policy, else decryption will fail. However, for a valid key time, the proxy-based server must be able to directly revoke a user using the SPIRC scheme and provide fine-grained access control.

Sequence Flow. Table 4 shows the sequence diagram for the access of the secure Mobile-based Healthfolder. The patient and health professionals personalise their device with credentials and identities on SE. After the HCE tap, they mutually authenticate each other and set a secure session key *Ksym*. The reader device requests to read or write to a $Section_i$. The card device sends the encrypted $Section_i$ along with a challenge *rei*. In the case of an emergency, the emergency professional obtains the Break the Glass (*BTG*) CP-ABE decryption keys *(KDRMabe,KDWMabe)* from the Healthsecure service. The health professional uses the read and write decryption keys to read and write to $Section_i$. After the session terminates, the patient mobile device sends the update for data sync to the digital vault. It also re-encrypts the Healthfolder with the new $Update_i$. After the session, the proxy-based server revokes the *BTG* CP-ABE decryption key for emer-

gency professional the SPIRC scheme.

5.3 Security Analysis

This section presents the security analysis for selective RBAC for Mobile-based Healthfolder.

S1: Confidentiality.
The mobile-based Healthfolder is encrypted by SPIRC and assures selective access by only authorised health professionals to assure confidentiality. SPIRC supports forward secrecy so that on revocation, a revoked user cannot access Healthfolder with his credentials.

S2: Selective read and write access.
Authorised stakeholders access various sections through selective RBAC. Each health professional has a separate CP-ABE decryption key to read and write to different sections and can access them only if the CP-ABE attributes associated with the key satisfies the corresponding access policy.

S3: Revocation.
The SPIRC scheme satisfies all revocation requirements for portable ciphertext *R1-R5* and provides flexibility to retain the secure Mobile-based Healthfolder on patient's mobile device. If an adversary $user_j$ finds Proxy data of another $user_i$, it will not help him with the decryption since each user has a different set of random constants maintained on the proxy-based server. There are however overheads of maintaining a constant set s_i for each $user_i$ on the proxy-based server. With scalable revocation, a patient can share health records across various hospitals and hence get mobility.

S4: Theft of device.
On the loss or theft of a registered device, the proxy-based server revokes the old credentials. Hence adversary cannot use the device. It issues new credentials along with the copy of re-encrypted Healthfolder on the new patient mobile device.

Hence it allows portability of secure health with direct sharing with trusted stakeholders.

S5: Emergency Break The Glass Key.
An emergency person authenticates with the Mobile-based Healthfolder and gets temporary CP-ABE read and write decryption keys from the HealthSecure service to provide emergency care. Later the proxy-based server revokes the emergency keys.

5.4 Implementation and Performance Analysis

Hardware Requirements: SPIRC-based CP-ABE requires around 40 MB of RAM and 1 GHz processor. This

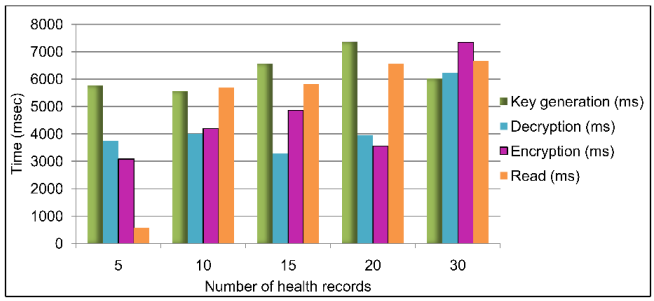


Figure 4: Impact of number of records

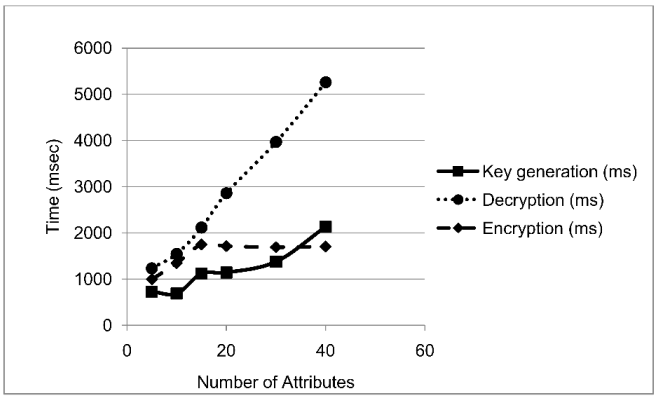


Figure 5: Impact of attributes on access time

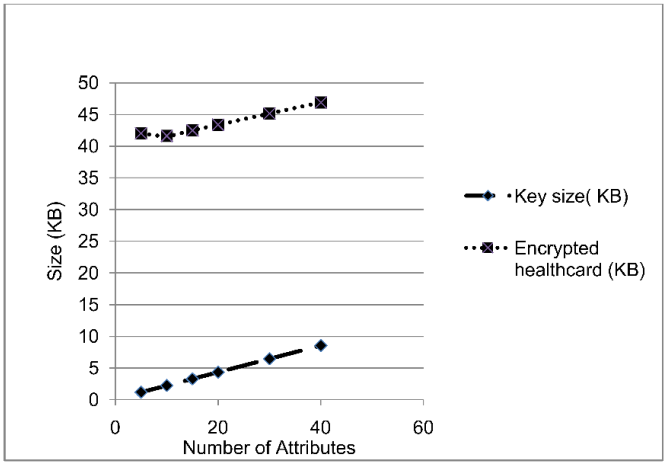


Figure 6: Impact of attributes on storage

Table 5: Average timings

Event	PIRATTE (ms)	SPIRC (ms)
Server Encryption	4197	4197
Proxy decryption	1864	450
Device decryption	2143	2143

Table 6: Comparison for revocation requirements

Requir.	PIRATTE [21]	M-PERMREV [13]	SPIRC (Proposed)
R1	No	Yes	Yes
R2	Yes	Yes	Yes
R3	Yes	Yes	Yes
R4	No	Yes	Yes
R5	Yes	No	Yes

configuration is already available in mid-range smartphones available in developing countries like India in the price range of 100-200 US dollars. CP-ABE has been implemented and tested successfully on Android-based mobile devices such as Samsung Galaxy Nexus device as well as IoT SBS (Single Board Computing) devices [12] such as Raspberry Pi and Intel Galileo.

The mobile-based healthfolder application discussed in the paper is based on NFC which is available currently in mid to high-end devices. NFC is primarily used for initial mutual authentication with the locality of reference as well as to automate pairing of Bluetooth. However, the application can also be deployed on the low-end devices without NFC, by using an alternate proximity technique of scanning secure QR-code using an inbuilt mobile camera to automate Bluetooth. Hence with the growing penetration of mobile devices across the world, the rollout of such a healthcare service in future can enable a rapid transition to health management.

The implementation of the Mobile-based Healthfolder comprises of a JavaScript Object Notation (JSON) file with a list of HL7 health records. The patient mobile device emulates an HCE-based card. A health professional accesses it directly using a reader application on the mobile device. Both mobile-based Healthfolder and the reader applications are implemented using:

- 2 Mid-range Android mobile device such as Sony Xperia M2 devices running Android 5.0.0 (Lollipop) which supports NFC-based HCE.
- Proxy-based CP-ABE scheme SPIRC scheme for selective access
- GO-Trust based secure microSD cards [14]. It includes Java card chip for SE on the microSD card, to store credentials and identities.
- Android SDK and Android Studio
- MongoDB and Python interpreter to maintain the HealthSecure service with Proxy-based Server for SPIRC.

The Health secure service outsources encryption to the proxy-based server. Decryption is performed partially on the user's mobile device and the Proxy-based server. When a patient visits an OPD for a department, typically a doctor reviews the previous health records. This paper assumes that a doctor would review approximate past 10 records at a time to gather the health history for a specific department. Hence, for the evaluation, 10 records are chosen, with an original size of 17KB and encrypted size of 57 KB. Table 5 shows the average timings for the Healthfolder encryption and decryption using PIRATTE and SPIRC scheme. It indicates that the overheads for the security computations for encryption and decryption of Healthfolder for both PIRATTE and SPIRC are similar with acceptable values for usage. SPIRC has lower overheads of proxy decryption as compared to PIRATTE since it associates proxy data with constant values instead of using Lagrange-based secret sharing in PIRATTE. Hence the total decryption time for SPIRC is lower as compared to PIRATTE.

Figure 4 shows the impact of the size of records for encryption, decryption and access time. It indicates that there is a significant increase in time to read as the number of records increase. However, it does not effect the encryption and decryption timings, since the size of ciphertext does not change. An AES key encrypts the Healthfolder, and the CP-ABE key is used to encrypt the AES key which remains constant. The read time comprises of communication time and decryption time to view the records. Since the communication time increases with the number of health records, the read time also increases. Figure 5 shows the increase in the timings for key generation, encryption and decryption with the increase in the number of attributes. Figure 6 illustrates that the increase in the number of attributes does not affect the storage size of encrypted Healthfolder. However, similar to CP-ABE, the key size increases with the increase in the number of attributes.

6 Performance Comparison

Table 6 illustrates the comparison of the different revocation techniques for the revocation requirements. Only SPIRC fulfils all the requirements *R1-R5*. Hence it is suitable for secure and selective access of a portable ciphertext and provides ease of usage to the owner and other non-revoked users.

Table 7 shows the comparison of CP-ABE techniques for overheads of storage and computational overheads of encryption and decryption. It also describes the terms used for comparison. The comparison assumes that all CP-

Table 7: Comparison of storage and performance

Scheme	CP-ABE [8]	PIRATTE [21]	M-PERMREV [13]	SPIRC (Proposed scheme)
Size of keys and ciphertext in different schemes				
PK	$2L_{G1} + L_{G2} + L_{GT}$	$2L_{G1} + L_{G2} + L_{GT}$	$2L_{G1} + L_{G2} + L_{GT}$	$2L_{G1} + L_{G2} + L_{GT}$
MK	$L_{G1} + L_{Zp}$	$L_{G1} + (1+t)L_{Zp}$	$L_{G1} + 2L_{Zp}$	$L_{G1} + L_{Zp}$
SK	$L_{G2} + (a + L_{G1} + L_{G2})\|A_U$	$L_{G2} + (a + L_{G1} + 2L_{G2})\|A_U$	$L_{G2} + (a + L_{G1} + L_{G2})\|A_U$	$L_{G2} + (a + L_{G1} + 2L_{G2})\|A_U$
CT	$(2\|A_C\| + 1)L_{G1} + L_{G2}$	$(2\|A_C\| + 1)L_{G1} + L_{G2}$	$(2\|A_C\| + 1)L_{G1} + L_{G2}$	$(2\|A_C\| + 1)L_{G1} + L_{G2}$
Broadcast	None	$tZ_p + \|A_U\|L_{G2} + Z_p$	$L_{G1} + L_{G2}$	$Z_p + \|A_U\|L_{G2}$
Comparison of computational overhead				
Encrypt.	$(2A_C + 1)G_1 + G_2$	$(2A_C + 1)G_1 + G_2$	$(2A_C + 1)G_1 + G_2$	$(2A_C + 1)G_1 + G_2$
Decrypt.	$2A_UC_e + (2\|S\| + 2)G_2$	$3A_UC_e + (2\|S\| + 2)G_2$	$2A_UC_e + (2\|S\| + 3)G_2$	$3A_UC_e + (2\|S\| + 2)G_2$

A_C: Attributes of ciphertext C; A_U: Attributes of user U; a: Length of an attribute; C_e: Number of bilinear pairings

G_i: Group or operations in group i, $i = 1$ or 2; S: Least interior nodes satisfying access structure (including root node); L*: Bit length of element in *; t number of users to be revoked

ABE schemes use asymmetric group pairing. All schemes have similar lengths for public key *PK*. However, the master key *MK* is shorter in SPIRC as compared to PIRATTE [21] since there is no generation of polynomial *P*. Both PIRATTE and SPIRC have similar lengths for private key *SK*, but which is longer as compared to the Bethencourt *et al.*'s CP-ABE [8] and M-PERMREV [13] schemes (both have same lengths for *SK*). *SK* is dependent on the number of attributes A_U allocated to the user *U*. The ciphertext length is dependent on the number of attributes of ciphertext A_C and is the same for all schemes. There is no broadcast overhead for Bethencourt *et al.*'s CP-ABE scheme [8]. The Broadcast overhead for PIRATTE is dependent on the number of revoked users and the number of attributes of a user A_U. The broadcast overhead of M-PERMREV is constant since it is only a state update for a user. However, it links a separate user state for each ciphertext and does not satisfy revocation requirement *R5*. SPIRC broadcasts a constant value for proxy data and is independent of the number of revoked users and dependent only on the number of attributes of a user A_U. Also unlike M-PERMREV, the proxy data is not be linked with the ciphertext such that there is an overhead of creating separate proxy data for each ciphertext for a user. The encryption time is dependent on the number of attributes in the ciphertext A_C and is similar to all schemes. The decryption time for PIRATE and SPIRC is higher as compared to the decryption time for CP-ABE and M-PERMREV, due to an extra bilinear pairing for proxy-base decryption. Due to simpler proxy data generated, the overall decryption time for SPIRC is smaller as compared to PIRATTE.

7 Conclusion and Future Work

Portable devices such as mobile devices can retain critical data encrypted with CP-ABE for fine-grained selective access control. This paper proposes a novel SPIRC scheme which improves PIRATTE [20] for scalable revocation. It satisfies all the revocation requirements *R1-R5* for ease of maintenance of ciphertext on a portable device. The overheads for generation of the master key and broadcast data as lower as compared to the PIRATTE scheme. Also, SPIRC does not associate the proxy data with the ciphertext as in the M-PERMREV scheme [13].

The paper presents a case study for using SPIRC for sharing secure portable Mobile-based Healthfolder with various health professionals over NFC as a contactless card. The Mobile-based Healthfolder is a next generation future Healthcard which can provide highly available and secure dispersed health records. The healthfolder can be shared with multiple health professionals using selective RBAC and provides mobility of patient across various hospitals. With the reduction in prices and the increase in penetration of mobile devices across the world, they can assist in the secure portable management of health data in emerging countries like India. The paper also successfully demonstrates the implementation and evaluation of a prototype of SPIRC for Mobile-based Healthfolder on mid-range Android devices with acceptable overheads for security and access.

Our work is the first novel attempt to address secure data on a portable device using Bethencourt *et al.*'s CP-ABE [8] with scalable user revocation. SPIRC can provide multi-user selective access to IoT devices such as users with different roles in a family access a car with their mobile devices to lock/unlock, configuration setup and access logs using selective RBAC.

The SPIRC scheme can be enhanced in future for supporting single key authority and multi-key authority delegation as well as attribute revocation. In future, we can compare SPIRC with other schemes such as those by Ibraimi *et al.* [19] using provably secure CP-ABE scheme [10] and by Lewko *et al.* [7] based on LSSS. Since mobile devices are vulnerable to security threats, the security scheme must assure that they are not susceptible to malware [1] and have trustful states. We can also use

secure smart card-based authentication [18] using Secure Element on a mobile device.

References

[1] A. Abdullah, Al-khatib, and A. W. Hammood, "Mobile malware and defending systems: Comparison study," *International Journal of Electronics and Information Engineering*, vol. 6, pp. 116–123, 2017.

[2] M. Alattar and M. Achemlal, "Host-based card emulation: development, security,and ecosystem impact analysis," in *Proceedings of IEEE High Performance Computing and Communications,* Aug. 2014.

[3] M. Ambrosin, M. Cont, and T. Dargahi, "On the feasibility of attribute-based encryption on smartphone devices," in *Proceedings of ACM IoT challenges in Mobile and Industrial Systems (IoY-Sys'15)*, pp. 49–54, 2015.

[4] N. Anciaux, M. Berthelot, L. Braconnier, *et al.*, "A tamper-resistant and portable healthcare folder," *International Journal of Telemedicine and Applications*, vol. 2008, 2008.

[5] N. Attrapadung and H. Imai, "Attribute-based encryption supporting direct/indirect revocation modes," in *Cryptography and Coding*, LNCS 5921, pp. 278–300, Springer, 2009.

[6] N. Aziz, N. Udzir, and R. Mahmod, "Extending TLS with mutual attestation for platform integrity assurance," *Journal of Communications*, vol. 9, pp. 63–72, 2014.

[7] A. Lewko, T. Okamoto, A. Sahai, *et al.*, "Fully secure functional encryption: Attribute-based encryption and (hierarchical) inner product encryption," in *Advances in Cryptology (EUROCRYPT'10)*, LNCS 6110, pp. 62–91, Springer, 2010.

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

[9] W. Chen, C. Hsu, Y. Lee, W. Jian, and H. Rau, "Developing electronic health records in taiwan," *IEEE IT Professional*, vol. 12, no. 2, pp. 17–25, 2010.

[10] L. Cheung and C. Newport, "Provably secure ciphertext policy ABE," in *Proceedings of the 14th ACM Conference on Computer and Communications Security (CCS'07)*, pp. 456Ũ465, 2007.

[11] V. Coskun, B. Ozdenizci, and K. Ok, "A survey on near field communication (nfc) technology," *Wireless Personal Communications*, vol. 71, pp. 2259–2294, 2013.

[12] A. Dmitrienko, Z. Hadzic, H. Löhr, A. Sadeghi, and M. Winandy, "On the feasibility of attribute-based encryption on internet of things devices," *IEEE Access*, vol. 36, pp. 25–35, 2016.

[13] S. Dolev, N. Gilboa, and M. Kopeetsky, "Permanent revocation in attribute based broadcast encryption," in *Proceedings of IEEE Cyber Security*, Dec. 2012.

[14] Go-trust, *Go-trustő Secure Microsd java*, Jan. 11, 2018. (`http://www.go-trust.com/products/microsd-java/`)

[15] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data," in *Proceedings of ACM Computer and Communications Security (CCS'06)*, pp. 89–98, Oct. 2006.

[16] HL7, *Introduction to HL7 Standards*, Jan. 11, 2018. (`http://www.hl7.org/implement/standards/`)

[17] J. Hur and D. K. Noh, "Attribute-based access control with efficient revocation in data outsourcing systems," *IEEE Transactions on Parallel and Distributed Systems*, vol. 22, pp. 1214Ũ1221, 2011.

[18] C. Y. Tsai, C. S. Pan, M. S. Hwang, "An improved password authentication scheme for smart card," in *Proceedings of International Conference on Intelligent and Interactive Systems and Applications*, Nov. 2016.

[19] L. Ibraimi, M. Petkovic, S. Nikova, P. Hartel, and W. Jonker, "Mediated ciphertext-policy attribute-based encryption and its application," in *Information Security Applications*, LNCS 5932, pp. 309–323, Springer, 2009.

[20] S. Jahid, P. Mittal, and N. Borisov, "Easier: Encryption-based access control in social networks with efficient revocation," in *Proceedings of the 6th ACM Symposium on Information, Computer and Communications Security*, pp. 411–415, 2011.

[21] S. Jahid, P. Mittal, and N. Borisov, ""PIRATTE: Proxy-based immediate revocation of attribute-based encryption"," in *arXiv preprint arXiv*, pp. 1–14, 2012.

[22] C. Lee, P. Chung, and M. Hwang, "A survey on attribute-based encryption schemes of access control in cloud environments," *International Journal of Network Security*, vol. 15, pp. 231–240, 2013.

[23] M. Li, Logan 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, pp. 131–143, 2012.

[24] C. Liu, W. Hsien, C. Yang, and M. Hwang, "A survey of attribute-based access control with user revocation in cloud data storage," *International Journal of Network Security*, vol. 18, pp. 900–916, 2016.

[25] J. Modi, M. Prajapati, A. Sharma, R. Ojha, and D. Jinwala, "A secure communication model for expressive access control using cp-abe," *International Journal of Network Security*, vol. 19, pp. 193–204, 2017.

[26] S. Narayan, M. Gagné, and R. Safavi-Naini, "Privacy preserving ehr system using attribute-based infrastructure," in *Proceedings of ACM Workshop on Cloud Computing Security Workshop*, pp. 47–52, 2010.

[27] Oracle, *Java Card Platform Security*, Jan. 11, 2018. (`http://www.oracle.com/technetwork/java/javacard/`)

[28] L. Pang, J. Yang, and Z. Jiang, "A survey of research progress and development tendency of attribute-based encryption," *The Scientific World Journal*, vol. 2014, 2014.

[29] D. Sethia, D. Gupta, and H. Saran, "Security framework for portable nfc mobile based health record system," in *Proceedings of the 12th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob'16)*, Oct. 2016.

[30] TCG, *Trusted Computing Group*, Jan. 11, 2018. (`https://www.trustedcomputinggroup.org/`)

[31] The National Archives, *U.k. National Health Service (NHS). Spine - NHS, Connecting for Health*, Jan. 11, 2018. (`http://www.connectingforhealth.nhs.uk`)

[32] Y. Tian, Y. Peng, G. Gao, and X. Peng, "Role-based access control for body area networks using attribute-based encryption in cloud storage," *International Journal of Network Security*, vol. 19, pp. 720–726, 2017.

[33] S. Y. Worcester, C. Wang, K. Ren, and W. Lou, "Attribute based data sharing with attribute revocation," in *Proceedings of IEEE 5th ACM Symposium on Information, Computer and Communications Security (ASIACCS'20)*, pp. 261–270, 2010.

Biography

Divyashikha Sethia Received B.Tech degree in Computer Science and Engineering from Maharaja Sayajirao University of Baroda in 1997 and M.Tech degree in Computer Science and Engineering from IIT Delhi in 2006. She has previously worked as a Software Engineer in the telecom industry in California, USA. She is currently an Assistant Professor in Department of Computer Science and Engineering at Delhi Technological University. Her research interests include distributed systems, computer networks, mobile computing and telemedicine

Huzur Saran Received B.Tech degree in Electrical Eng., Indian Institute of Technology, Delhi in 1983 and PhD in the Computer Science University of California, Berkeley in 1990. Dr Saran has also led many other entrepreneurial technology development efforts such as Solidcore Inc., which was purchased by Mcafee for their Dynamic Whitelisting technology developed entirely in India under Prof Saranąęs Leadership. He is currently professor of Computer Science at IIT Delhi and Heads the Amar Nath and Shashi Khosla School of Information Technology, at IIT Delhi. His research interests include Computer Systems, Security and Algorithms.

Daya Gupta received her M.Sc.(Computer Science) at I.I.T Delhi, and PhD in Computer Science from the University of Delhi. She is Professor in Department of Computer Science and Engineering at Delhi Technological University. She is a Senior Member, I.E.E.E. Her research interests include Information Security, Requirements Engineering, Swarm Intelligence, Nature Inspired Algorithm, Data Warehouse and Data Mining.

Data Verification Using Block Level Batch Auditing on Multi-cloud Server

G. L. Prakash, Manish Prateek, and Inder Singh
(Corresponding author: G. L. Prakash)

School of Computer Science and Engineering, University of Petroleum and Energy Studies
Bidholi, Via Prem Nagar, Dehradun, Uttarakhand 248007, India
(Email: prakashgl@ddn.upes.ac.in)
(Received Mar. 20, 2017; revised and accepted June 26, 2017)

Abstract

In cloud computing, storage as a service provides an on-demand, flexible data sharing across the networks. This reduces the burden of local data storage management and avoidance of resource maintenance (Hardware or software). In this paradigm, data owner loses the control of the outsourced data, once the data leaves the data owner premises. Due to this, the data on an untrusted cloud server is at risk in terms of integrity, confidentiality and availability of the outsourced data. In order to maintain the outsourced data without corruption from the internal or external adversary, an efficient data auditing verification method is required for data verification. In this paper, we propose a flexible data auditing method using block level auditing of data distributed on multiple cloud servers. This method utilizes the Computational Diffie-Hellman (CDH) and Decisional Diffie-Hellman (DDH) problem solving techniques. The performance of data verification with different sizes of data blocks on multiple servers. Compared to the existing methods of data auditing the proposed method minimizes the computation, communication and storage overheads.

Keywords: Auditing; Cloud Computing; Corruption; Storage management; Verification

1 Introduction

In modern computing technology, cloud-computing paradigm is an important technology used to provide various remote services such as, computing, storage, memory and other services with reduced computing cost when compared with many traditional approaches. There are various cloud service providers available in recent days including Amazon Web Services, Microsoft Azure, Google Cloud Platform and IBM Cloud that provide storage as a service [5]. Storage as a Cloud service is one of the important features of cloud computing used to share user's data across the network [8,14]. The cloud servers examine the outsourced data very frequently because the data can be lost or corrupted due to hardware failure, software failure or from the assailants [12,17].

Maintaining the integrity of outsourced data in a cloud server is an important issue in cloud computing [19]. In order to maintain the reputation of the cloud service, the cloud service providers set access restrictions on the services it provides to the users [2]. To avoid losing of profit of the service and to maintain the quality of the cloud service, verification of the integrity of outsourced data becomes mandatory before data utilization. To verify the integrity of outsourced data, various traditional approaches such as RSA, hash functions, MAC, Digital signature [9–11] are proposed. These proposed methods retrieve the entire data from the servers to verify the correctness of the outsourced data so that the auditor can derive the user data from this information and it takes more computation and communication cost, which can degrade the efficiency of the system. Therefore, the traditional integrity checking approaches are not suitable in cloud computing to utilize the resources optimally [6]. In general, the size of the data is very large for downloading the server and verification of data integrity would demand availability of more resources.

There are many methods proposed for checking data integrity without downloading the entire data from the server. This verification can be either private verification, public verification or delegated verification also called as public auditing. In these methods, it is essential to divide data into smaller blocks. Random verification of data blocks is preferred instead of retrieving the entire data block for verification after the data owner had signed these data blocks.

Organisation of the rest of the paper is as follows; Section 2 explains the various existing data auditing methods. Sections 3 and 4 explains the statement of the problem and the detailed algorithms for the method proposed. Section 5 presents the performance analysis of the proposed method and finally we concluded in the Section 6.

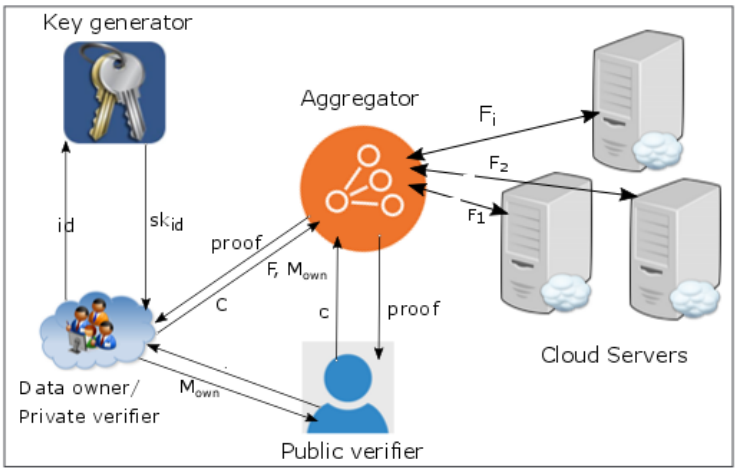


Figure 1: Data auditing system model

2 Related Work

In cloud computing, to ensure the integrity of the data on untrusted storage on single cloud the various public and private auditing schemes are proposed [1, 3, 4].

In Ateniese *et al.* [1], proposed a public auditing model as Provable Data Possession (PDP) for auditing data on untrusted storage entity. They introduced homomorphic linear authenticators to audit the selected outsourced data blocks of the outsourced file. This method may leak the user information to the auditor during auditing process and which may leads to helps to derive the user information, therefore it does not provide the security for the outsourced data.

In Juels *et al.* [4], introduced a Proof of Retrievability (PoR) model for verification and retrieval of data from remote data storage service using error-correcting codes. This method has the following drawbacks: 1) It has fixed data auditing challenges, 2) Suffers from public and delegate verification.

In Yujue *et al.* [20], addressed identity-based data outsourcing for distributed users. In this method of audit, the data users have to authorize the dedicated proxy before storing data on cloud server and which is more controlled way of outsourcing data on cloud server. To verify the integrity of outsourced data is more expensive.

In [7, 13, 15, 16] proposed a data auditing protocol on the cloud server to support the batch auditing on multi-servers. In these methods, individuals used data tags to the owner and these cannot help to combine multi-owner tags to conduct batch auditing. To combine these individual tags, third party auditor is introduced which takes additional computation and communication cost. Due to these overhead this method reduces the efficiency of the auditing system.

In [21, 22] proposed the data privacy protocol for auditing user's data in cloud storage server by using the bilinear privacy operations to verify the correctness of the response message. The drawback of this method is that, for multi-cloud auditing to segregate the data blocks of the multiple users, the auditor takes more computational task and which is a low end user entity in the cloud storage system. This method suffers from yet another drawback while using unencrypted used information for auditing process, thereby empowering the auditor to derive user data.

3 Statement of the Problem

3.1 System Model

The system model considered in the proposed work as shown in the Figure 1. It consists of five components such as; *key generator, cloud servers, verifier, cloud users and aggregators.*

Key generator: It is an entity, which receives the identity of the user(ID) and generate the secrete key(*sk_id*) for the user(ID) using a computational Diffie-Hellman (CDH) method.

Cloud users: It is an individual user or an organization which outsource the data on multi cloud servers for maintenance and management of shared data.

Verifier: It is an entity, either the data user or third party auditor to check the correctness of outsourced data using Decisional Diffie-Hellman method.

Cloud servers: It is a set of servers, which managed by the cloud service provider to provide the storage service, which has massive compute and storage facility.

Aggregator: It is an independent and trusted entity. Which distribute the requests to the servers and aggregate the responses from the servers.

3.2 Security Model

In cloud data storage model, the auditor and the cloud servers are semi trusted entities, which means they are

honest but it is curious about the received data. Due to this semi trusted entities this may create the following attacks.

User attack: Poor identity and access management procedures an unauthorized intruder can attack the user

Data segregation: Incomplete security perimeters and configuration of virtual machines could provide a threat against data integrity.

Response attack: The semi trusted server may generate the response message for the requested data blocks from the previous audits without using the actual owner's data.

Data attack: The server and auditor can derive the user's data using metadata information in the frequently auditing task.

3.3 Objectives

In the proposed method the following objectives are achieved for auditing outsourced data on cloud storage server.

In our proposed method we are achieved the following objectives for auditing data on multi cloud storage.

Flexible data auditing: The private or public data verification method can be applied based on the priority of the outsourced data.

Privacy of the data: In either of the auditing method, the verifier cannot derive the user data from the metadata.

Lightweight overhead: To optimize the storage, computation and communication overhead to perform the data auditing on cloud server.

3.4 Notations

The various symbols are used in this paper is listed in Table 1 as follows.

3.5 Cyclic Group Operation

Consider G1, G2, G_T are cyclic multiplicative groups of order prime number p, bilinear map using these groups is defined as $G1 \times G2 \longrightarrow G_T$ of order p.

The property of the map e is that for all a, $b \in p$ is defined as that $e(g_1^a, g_2^b) = e(g_1, g_2)^{ab}$ for all $g_1 < G_1$ and $g_2 < G_2$. The group G_T is distinct from input groups G_1 or G_2 and all the elements of this map is also elements of group G_T.

Table 1: Notations

Symbol	Meaning
F	erasure encoded file $F = \{F_{ij}\}$
t	file tag
n	number of data blocks
s	number of sectors per block
σ	block authenticator
M_{owner}, M_{agg}	File metadata for owner and aggregator
$H(), h()$	hash functions
Φ	set of authenticators
r and x	random number
c	challenge message
CS_{l_i}	set of cloud servers
p	prime number
msk_{id}	master secret secret key

4 Data Auditing Design Methodology

4.1 Basic Auditing Method

The basic data integrity verification scheme consists of three stages such as; key generation, metadata generation and data auditing. The detail of these stages has shown in Figure 2.

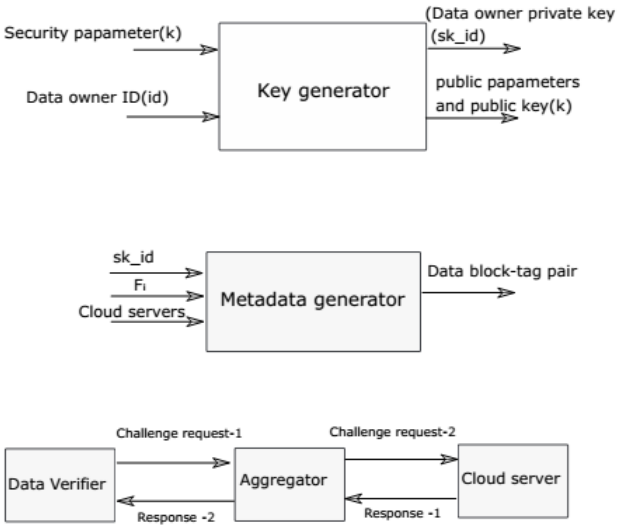


Figure 2: Basic auditing model

4.1.1 Key Generation

It is an entity, which takes the input as security parameter(k) and the data owner identifier (id) and generates the public parameters, secrete key, public key and owners private key(pk_id).

4.1.2 Metadata Generation

To generate the metadata for a given file (F_i) of the owner (id), the metadata generator takes input as owners private key (pk_id) and the file (F_i) as input and generates the signature of file blocks interns of $block - tag$ pair.

4.1.3 Data Auditing

The data auditing process consists of five steps of request, response and verification among verifier, aggregator and cloud servers.

1) Verifier send the challenge request to aggregator for verification of selected number of data blocks stored on cloud servers.

2) The aggregator searches the requested data blocks meta data from the metadata table and then distribute the request to the corresponding cloud server.

3) After receiving the responses from the cloud servers, the aggregator combines all the responses.

4) Aggregator sends the final combined response to the verifier.

5) The verifier verify the response message using bilinear map operation. If the response is valid, verifier confirms data blocks are not modified, otherwise he declares data blocks are modified.

4.2 Proposed Batch Auditing Method

The proposed data auditing method consists of key generation, metadata generation and data verification procedure. The detailed explanation of these procedure as follows.

4.2.1 Key Generator

The key generator select two random positive integer numbers r and x and calculate $A = g^x$ and $B = g^r$ where g is the generator i.e $g <$ group G_1, keeps x has secrete key and $\{g, A\}$ as public parameters.

For the given data owner (id) key generator calculate the signature using Equation (1) and sends the private key $sk_{id} = (\sigma and B)$ to the data owner. Then the data owner verifies the private key using Equation (2).

$$\sigma = r + x(H(id, B)) \bmod q \quad (1)$$

$$g^{\sigma} \stackrel{?}{=} BA^{H(id,B)} \quad (2)$$

The detailed algorithm for key generation and authorization is explained in Algorithm 1. The proof for the correctness equation and example as follows;

$$g^{\sigma} \stackrel{?}{=} BA^{H(id,B)}$$

$$\begin{aligned} LHS &= g^{\sigma} \\ &= g^{r+xH(id,B)} \\ &= g^{r}.g^{xH(id,B)} \\ &= BA^{H(id,B)} \\ &= RHS. \end{aligned}$$

For example, $g = 3$, $q = 11$, $r = 3$, $x = 4$:

$$\begin{aligned} A &= g^x = 3^4 \\ B &= g^r = 3^3 \\ \sigma &= r + x(H(id, B)) \bmod q \\ &= 3 + 4.H(id.B) mod 11 \\ g^{\sigma} &\stackrel{?}{=} BA^{H(id,B)} \\ 3^{3+4.H(id.3^3) mod 11} &= 3^3.3^{4.H(id.3^3) \bmod 11} \\ &= 3^{3+4.H(id.3^3) \bmod 11} \end{aligned}$$

Algorithm 1 Key Generation

input: User identity (id)
output: Master secrete key (x), Public parameters (p, q, g, A, H)

1: Select a random number x, r from a set of positive integer numbers Z_q^*
2: compute A, B and σ
$A = g^x$ and $B = g^r$ $\sigma = x + r(H(id, B)) \bmod q$
3: keeps x has master secrete key.
4: sends private key $sk_{id} = (B, \sigma)$ and public parameters to user.
5: Verify the user identity id by solving the DDH problem $g^{\sigma} \stackrel{?}{=} BA^{H(id,B)}$
6: If the equation is verifies then accept the user (id) private key sk_{id}, otherwise reject it.

4.2.2 Metadata Generation

The data owner with the valid private key (sk_{id}) prepares the metadata for the file F and stores the metadata and the corresponding file on cloud server CS. Consider the file F is split in to n blocks and each block of s sectors i.e; $F = F'_{ij}$, where $i = 1ton$, $j = 1tos$ and F' is the encrypted data block. The Data owner calculates the hash value for each sector using MD5 algorithm i.e. $h1(F'_{ij})$ and prepare the metadata M_i for the file block F_i using Equation (3). Then the owner sends the file blocks and metadata $\{F_i$ and $M_i\}$ to the cloud server CS_{l_i}. The procedure of metadata generation is explained in Algorithm 2.

$$M_i = (h(CS_{l_i}, i, name_i).\Pi_{j=1}^{s} u_{ij}^{F_{ij}})^{\sigma} \quad (3)$$

Where name i is the identifier of each block, j is the sector number in the data block and u_{ij} is the random number.

Algorithm 2 M_i

input: File (F), sk_{id}
output: Metadata M_i for the file block F_i

1: Data owner split the file F in to n blocks $\{F_i\}$, and each encrypted data block in to s sectors i.e., $\{F'_{ij}\}$ where $i \leq n$, and $j \leq s$
2: Data owner selects s random number vector $\{u_i\}$ where $j \leq s$
3: calculate the hash values for each encrypted file block i.e., $F_{ij} = h(F'_{ij})$
4: calculate the metadata for the ith file block i.e, $M_i = (h(CS_{l_i}, i, name_i).\Pi_{j=1}^{s} u_{ij}^{F_{ij}})^{\sigma}$
5: Data owner adds $\phi_i = (i, u, CS_{l_i}, name_i)$ to the M_i table
6: Data owner sends M_i to aggregator, then the aggregator stores in his metadata table, M_{agg})

and stores file blocks in cloud server CS_{l_i}

4.2.3 Data Verification

To verify the data stored in cloud server CS, the auditor sends a request for selected number of blocks c to aggregator and aggregator identifies the corresponding cloud server using the metadata table M_{agg} and further sends a request to the corresponding cloud server CS_{l_i}. After receiving the request from the aggregator, cloud server prepares the response message and send to the aggregator. The aggregator combines all the responses and sends the aggregated response to the auditor. The auditor verifies the correctness of the response message using the Equation (4). The details of the data auditing algorithm is explained in Algorithm 3.

The block diagram for private and public data verification as shown in the Figure 3 and Figure 4 respectively.

$$e(M, g) \stackrel{?}{=} e(\Pi_{i=1}^{c} h_i^{F_i} \Pi_{j=1}^{s} u_j, BA^{H(id,B)}) \quad (4)$$

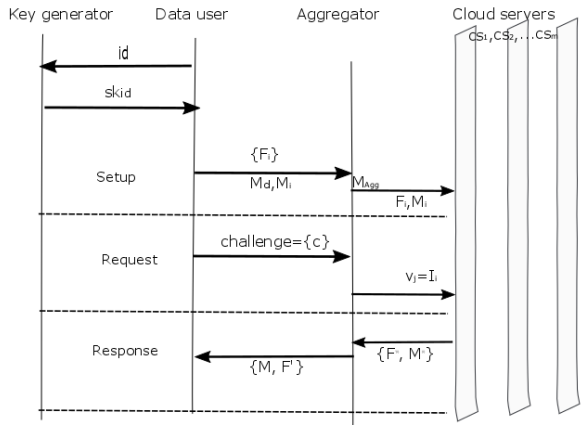


Figure 3: Batch auditing setup and private verification

The proof for data auditing response verification as follows.

$$\begin{aligned} LHS &= e(M, g) \\ &= e(\prod_{cs_i} M^{(i)}, g) \\ &= e(\prod_{cs_i} \prod_{v_l \in M_i} M_{v_l}, g) \\ &= e(\prod_{cs_i} \prod_{v_l \in M_i} h_l \prod_{j=1}^{s} u_j^{F_{v_l j}}, g^{\sigma}) \\ &= e(\prod_{i=1}^{c} h_i \prod_{j=1}^{s} u_j^{F_{v_l j}}, BA^{H(id,B)}) \\ &= RHS \end{aligned}$$

Algorithm 3 Data_Audit(c)

input: Challenge message c data blocks
output: Metadata for the file block F_i

1: The private or public verifier sends the challenge message (c) to the aggregator
2: Aggregator prepares the index set(I_i) for the corresponding c request blocks using $block - tag$ pairs stored in cloud server.
3: Aggregator sends the index set(I_i to the cloud servers.
4: for each cloud server v_l calculates $(M^{(i)}, F_j^{(i)})$ and send to the aggregator.
$M^{(i)} = \Pi_{v_l \in I_i}\{M_{v_l,j}\}$
$F_j^{(i)}) = \Sigma_{v_l \in I_i}\{F_{v_l,j}\}$
5: Aggregstor prepsres the aggregated message (M and F') then sends to the verifier.
$M = \Pi_{cs_i} M^{(i)}$ and $F'_l = \Sigma_{cs_i} F_l^{'(i)}$
6: Verifier solves the following DDH problem and returns the status of the file block.
$e(M, g) \stackrel{?}{=} e(\Pi_{i=1}^{c} h_i \Pi_{j=1}^{s} u_j, BA^{H(id,B)})$
7: If the above equation holds then it responds success, otherwise it responds failure.

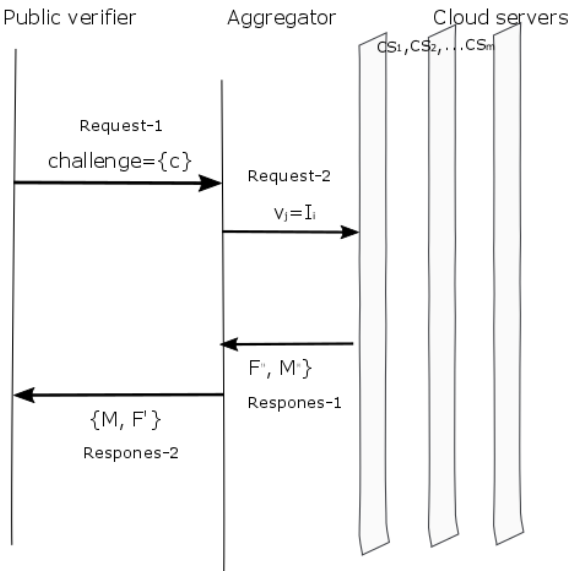


Figure 4: Data auditing using public verification

Table 2: Tag generation time for different file sizes

File Size(MB)	Tag generation Time(Seconds)
1	1.87
2	3.17
4	4.27
6	6.85
8	10.05
10	14.65
12	18.59
14	19.87

5 Performance Analysis

5.1 Simulation Setup

To evaluate the performance of our proposed data auditing method the following simulation setup are considered. All the algorithms are implemented using Python programming language with built in cryptographic functions in Python library. The simulation result is tested on Amazon Web Service virtual machines (VM).

Two VMs is run as a cloud server and one VM run as a aggregator with Ubuntu operating system, t2.small vCPU and 2GB EBS storage configuration. The data owner and the public verifier runs on a Laptop with 64 bits Windows 10 operating system, i3 intel processor and 4GB physical memory configuration.

5.2 Result Analysis

In ordered to show the performance of the proposed method, we compared with existing ID-DPDP [18] data auditing method. The performance of the algorithm is evaluated based on the tag generation, tag verification cost and file auditing cost for different data block size with different batch size.

Table 2, shows the comparison of tag generation cost for different sizes of files with 256KB of data blocks size. First column represent the different block size in Megha Bytes and Second column represents the tag generation cost in seconds. It is observed that, for smaller sized file sizes tag generation cost is linear than the larger file sizes. The proposed method shoes that, it has lightweight computation overheads for the larger file sizes. Table 3 shows the tag generation cost for different sizes of the data blocks of 1MB file size.

Table 5, shows the comparison of tag verification cost on cloud servers. To verify the different sizes of data block for 1MB file tag verification cost shows that, for smaller size of data blocks both the methods has same order of growth for tag verification cost. In case of larger size of data blocks our proposed method has better performance than the ID-DPDP method.

In Table 5, shows that, the performance comparison to audit 10MB data file with 256KB of data blocks for different number of batch size. For smaller batch size of auditing both the method has same order of auditing cost and for larger batch size our proposed method performance is better than the existing method.

Table 3: Tag generation time for different data block sizes

Data block Size(KB)	Tag generation Time(Seconds)
256	1.856
500	1.047
768	0.677
1024	0.384

Table 4: Tag verification cost with different block size

Data block Size(KB)	Tag generation Time(Seconds)
256	0.0856
500	0.0447
768	0.0327
1024	0.0154

The cost performance comparison for auditing data on multiple servers interms of computation, communication and storage overheads with Zhu, ID-DPDP as shown in Table 6,

6 Conclusions

In this paper, the flexible data verification method for outsourced data integrity checking is proposed. This introduce the data aggregator, which distributes the requests and aggregates the response from the servers during the auditing process. The proposed method utilizes the CDH and DDH problem to verify the correctness of the outsourced data. Finally, the performance comparison of the proposed data verification method with the existing method. The experimental results and security analysis shows that, the proposed method is better than the existing method

References

[1] G. Ateniese, R. Burns, R. Curtmola, *et al.*, "Provable data possession at untrusted stores," in *Proceedings*

Table 5: Tag verification cost with different block size

Number of Blocks	Data Verification Time (Seconds)
4	3.1856
8	5.0127
12	8.5327
24	10.0154

Table 6: Computation performance of auditing methods

Auditing Method	Computation overhead	Communication overhead	storage overhead
Zhu [23]	expo:(2s+ns+2n+c) mul:(c+s+2) pairing: 3	$c.log_2 n + (1+s)log_2 q$	$T_{cl} + T_o$
ID-DPDP [18]	expo:(2s+ns+2n+c) mul:(sn+s+1), hash:(ns+cs) pairing: 2	$log_2 n + (2+s)log_2 q$	$T_{cl} + T_o$
Proposed method	expo:(2ns+2n+c) mul:(sn), hash:(ns+cs) pairing: 3	$log_2 n + s log_2 q$	$M_{owner} + M_{Aggre}$

of the 14th ACM Conference on Computer and Communications Security (CCS'07), pp. 598–609, 2007.

[2] B. Balusamy, P. V. Krishna, G. S. T. Arasi, and V. Chang, "A secured access control technique for cloud computing environment using attribute based hierarchical structure and token granting system," *International Journal of Network Security*, vol. 19, no. 4, pp. 559–572, 2017.

[3] J. Han, H. Yan, J. Li and Y. Zhang, "A novel efficient remote data possession checking protocol in cloud storage," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 1, pp. 78–88, 2017.

[4] A. Juels, Jr. Kaliski, S. Burton, "Pors: Proofs of retrievability for large files," in *Proceedings of the 14th ACM Conference on Computer and Communications Security*, pp. 584–597, 2007.

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

[6] S. K. Madria, "Security and risk assessment in the cloud," *IEEE Compurter*, vol. 49, no. 9, pp. 110–113, Sept. 2016.

[7] J. Mao, J. Zhang, P. Li, "An oriented-group supporting multi-user public auditing for data sharing," in *IEEE International Conference on Smart City/SocialCom/SustainCom*, pp. 592–599, 2015.

[8] A. Mosa, H. M. El-Bakry, S. M. Abd El-Razek, S. Q. Hasan, "A proposed E-government framework based on cloud service architecture," *International Journal of Electronics and Information Engineering*, vol. 5, no. 2, pp. 93–104, 2016.

[9] Z. Ren, L. Wang, Q. Wang, M. Xu, "Dynamic proofs of retrievability for coded cloud storage systems," *IEEE Transactions on Services Computing*, 2016.

[10] R. L. Rivest, *The MD5 Message-digest Algorithm*, Apr. 1992. (`https://tools.ietf.org/html/rfc1321`)

[11] R. L. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," *Communications of ACM*, vol. 21, pp. 120–126, Feb. 1978.

[12] D. Song, I. Fischer and U. Shankar, "Cloud data protection for masses," *IEEE Computing*, vol. 45, no. 1, pp. 39–45, 2012.

[13] S. Tan, L. Tan, X. Li, Y. Yan, "An efficient method for checking the integrity of data in the cloud," *China Communications*, vol. 11, no. 9, pp. 68–81, Sept. 2014.

[14] Y. Tian, Y. Peng, G. Gao, X. Peng, "Role-based access control for body area networks using attribute-based encryption in cloud storage," *International Journal of Network Security*, vol. 19, no. 5, pp. 720–726, 2017.

[15] B. Wang, B. Li, and H. Li, "Panda: Public auditing for shared data with efficient user revocation in the cloud," *IEEE Transactions on Services Computing*, vol. 8, no. 1, pp. 92–106, 2015.

[16] B. Wang, H. Li, X. Liu, F. Li, X. Li, "Efficient public verification on the integrity of multi-owner data in the cloud," *Journal of Communications and Networks*, vol. 16, no. 6, pp. 592–599, 2014.

[17] C. Wang, K. Ren and Q. Wang, "Security challenges for public cloud," *IEEE Internet Computing*, vol. 16, no. 1, pp. 69–73, 2012.

[18] H. Wang, "Identity-based distributed provable data possession in multi-cloud storage," *IEEE Transactions on Service Computing*, vol. 8, no. 2, pp. 328–340, 2015.

[19] Q. Wang, C. Wang, K. Ren, *et al.*, "Enabling public auditability and data dynamics for storage security in cloud computing," *IEEE Transactions on Parallel and Distributed Systems*, vol. 22, no. 5, pp. 847–859, May 2011.

[20] Y. Wang, Q. Wu, B. Qin, *et al.*, "Identity-based data outsourcing with comprehensive auditing in clouds," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 4, pp. 940–952, 2017.

[21] L. Xia, L. Yang, "An efficient and secure public batch auditing protocol for dynamic cloud storage data," in *IEEE International Computer Symposium*, pp. 671–675, 2016.

[22] K. Yang and X. Jia, "An efficient and secure dynamic auditing protocol for data storage in cloud computing," *IEEE Transactions on Parallel and Distributed Systems*, vol. 24, no. 9, pp. 1717–1726, 2013.

[23] Y. Zhu, H. Hu, G. J. Ahn and M. Yu, "Cooperative provable data possession for integrity verification in multicloud storage," *IEEE Transations on Parallen and Distributed Systems*, vol. 23, no. 12, pp. 2231–2244, 2012.

Biography

Prakash G L is an Assistant Professor with the Department of Computer Science and Engineering of University of Petroleum and Energy Studies, Dehradun, India. He received his B.E and M.E degrees in Computer Science and Engineering from Bangalore University, Bangalore. He is presently pursuing his Ph.D program in the area of data storage and security in Cloud Computing from University of Petroleum and Energy Studies. He has published more than five international journals and ten international conference papers.

Dr. Manish Prateek did his Undergrad and Post Grad degree in the field of Computer Science from South West State University, (formerly known as Kursk State Technical University), Russia, in 1996. Since then he worked at different level in the IT industry in India as well as in Middle East. He did his PhD in the area of Manufacturing & Robotics and was awarded the degree in the year 2007. In 2005 he took up his career into technical education and started as Associate Professor and Head, Dept. of Information Technology at GRIET Hyderabad and later became a full Professor at the age of 36. Currently he is working as Professor & Associate Dean at UPES, Dehradun. His area of research includes Robotics and Automation, Image Processing and Patter Recognition, Cyber Security, Machine Vision etc. So far, more than 40 research papers have been published by him in different International Journals. He is also a recipient of Lifetime Achievement Award for his contribution in the field of education and research by the prestigious scientific society Pentagram Research Centre. He is also a member at the prestigious International Federation for Systems Research (IFSR), Austria

Dr. Inder Singh, M.Sc.(IT), M. Tech. (IT), Ph.D., Microsoft Certified Professional, IBM DB2 certified, e-commerce certification from Asset International, Dell EMC's Certified Data Science Associate. He is an Assistant Professor (S.G.) at School of Computer Science and Engineering, UPES, Dehradun. He has over 16 years of working experience. He has started his career as Systems Administrator and switched to teaching profession after 6 years. His area of research includes Computer Networks, Cloud Computing and Virtualization, and Data Science. So far, he has published and presented more than 18 research papers in different International Journals and conferences.

A New Access Control System Based on CP-ABE in Named Data Networking

Tao Feng, Jiaqi Guo
(Corresponding author: Jiaqi Guo)
School of Computer and Telecommunications, Lanzhou University of Technology
287, Lanping Road, Qilihe District, Lanzhou 730050, China
(Email: flyingfairy@163.com)
(Received Mar. 22, 2017; revised and accepted June 26 & July 29, 2017)

Abstract

Named Data Networking (NDN) paradigm introduces a novel security communication model where any router in the network can store mass data, which is convenient for consumers to obtain data from any nearby router. However, such a radical change causes new challenges for NDN access control since the data publisher loses control over the published data. In most of the existing access control mechanisms in NDN, publishers are required to be always online to authenticate consumers, also revocation and user privacy is considered scarcely. Focusing on those problems, the work of this paper is proposing a new access control system in NDN, which comprises the framework and algorithm, security definition and proof. The basic of this paper is a decentralized ciphertext-policy attribute based encryption (CP-ABE) scheme under prime order groups, which solves the above problems effectively. In this new scheme, indirect revocation can be combined with the in-network storage technique in NDN properly. Privacy is guaranteed by using the partially-hidden access structure to realize recipient anonymity. Finally, it is proved to be static security.

Keywords: Access Control; CP-ABE; NDN; Privacy-reserving; Revocable

1 Introduction

In current Internet, consumers are mainly interest in accessing and consuming content, irrespective of where the content comes from and who publishes the content. It is difficult for the traditional TCP/IP network to meet the new requirements, such as supporting massive content distribution, mobility, security and so on. Therefore, researchers consider changing the network architecture radically. Information Centre Networking (ICN) [30] is a candidate of future Internet which employs a content-oriented mechanism instead of host-oriented mechanism. NDN [31] is an emerging ICN project, as the same, it treats content as core element. NDN relies on in-network storage which makes the same content distribute to multiple locations in the network, so it is convenient for consumers to get content from any neighbor router.

Considering the particular architecture of NDN, especially the in-network storage, access control is very important for the security in NDN. Once data is published in the network, it will be stored in any router where it passes by. So the publishers face the risk of losing control of published data. Therefore, more and more researchers pay attention to access control in NDN [28].

However, most existing NDN access control schemes need publishers always online to authenticate consumers, which is impractical and leads to bad effects on in-network storage of routers. What's more, in these schemes, revocation is not considered and even the publisher-always-online problem is not resolved. Using Attribute-based Encryption (ABE) can overcome this problem [2, 32]. However, a trusted central authority (CA) is needed in such schemes, which has all the keys of the system, it is prejudicial to system security. In addition, the access structure included in ciphertext also reveal the sensitive information of users.

To address the problems above, a new access control model in NDN is constructed and an efficient NDN access control scheme is proposed. This scheme is decentralized, revocable and privacy-preserving without breaking the in-network storage mechanism of NDN. For publishers, they can develop the access policy by themselves; and for consumers, they can easily get the data from nearby routers; for both of them, their sensitive privacy information included in access structure will be hidden. In general, the system proposed in this work has the following goals:

Decentralization. A secure access control system in NDN is presented based on a CP-ABE scheme [23] which has multiple attribute authorities (AAs) but without a central authority (CA), so that it is more practical. Also, the scheme is on the base of prime order groups with high efficiency.

Revocability. An indirect revocation approach is proposed for the above scheme based on periodically updating attribute-based keys of the non-revoked consumers. Both backward and forward security is achieved.

Privacy-preserving. Anonymous access structures are used to make the sensitive information hidden, namely the attribute values are hidden. So this scheme realizes recipient anonymity and privacy-preserving.

The structure of this work arranged as follows. In Section 2 are some works about access control in NDN and ABE schemes. Some preliminaries are presented in Section 3. The system model, security requirements, security assumptions and security game are all given in Section 4. The formal algorithms of this scheme are described in Section 5. Sections 6 and 7 analyzes security and performance respectively. Finally, conclusion and prospect are in Section 8.

2 Related Works

2.1 Access Control in NDN

Differ from the IP network, in NDN, requesting and publishing content are both based on content name rather than IP address. In order to retrieval content effectively, there are two kinds of packets in NDN as shown in Figure 1. The interest packet carries the requirement of the consumer, while the data packet carries content.

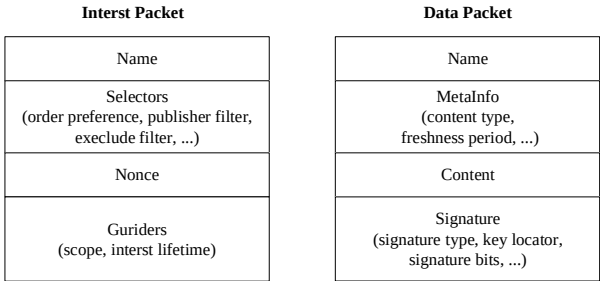


Figure 1: Packets in NDN architecture

In NDN, the publisher signs a packet for its integrity and authenticity, but fine-grained access control is also required. Thanks to in-network storage, routers can satisfy corresponding requirements in interest packets irrespective of whether the interests are from authorized or unauthorized consumers. So access control in NDN cannot be implemented by a single entity. Therefore, Jacobson *et al.* [31] mentioned that access control in NDN can be achieved based on encryption. Data encryption is an effective way of access control, but the traditional encryption cannot be used in NDN efficiently, since the number of consumer increases, and the encrypted copies of each data will also increase proportionally.

The existing access control schemes in NDN are showed in Table 1. In schemes [7, 9, 11, 26, 29], an always online publisher is required for authenticating consumers, but it is impractical and affects the in-network storage mechanism in NDN. Revocation is not considered in schemes [9, 11, 17, 29], and in schemes [25], although it solves the problem of user revocation, a proxy server is also required to be always online, and the formal algorithm and security proof are not given. The access control mechanisms without effective revocation can't be used in practice. An effective revocable access control scheme accommodates the in-network storage mechanism in NDN has not been reported.

2.2 ABE Schemes

In 2005, the concept of ABE first appeared in the scheme of fuzzy identity-based encryption [24]. Using ABE, publishers can share data with specific consumers without their public keys and identities [27]. Two more practical ABE schemes were constructed on the first ABE scheme, key-policy ABE (KP-ABE) [10, 19] and ciphertext-policy ABE (CP-ABE) [3,14]. This paper considers publisher developing and performing access control policy, so CP-ABE is used because the access structure is included in ciphertext. Recently, the original ABE has been improved, and various practical ABE schemes have been proposed which can achieve multi-AA, revocation, hidden access structure and so on.

Chase *et al.* proposed the first multi-authority ABE scheme [5], then some multi-authority CP-ABE schemes had been presented [16, 22], but the CA in these schemes has all the keys of the system. Then Chase and Chow [6] presented a scheme where CA is not needed, but it is only constructed on AND structure. Subsequently, a fully decentralized CP-ABE scheme without CA [15] is presented in 2011. In 2015, an improved large-scale decentralized multi-authority CP-ABE scheme is proposed [23], which is based on prime order groups and it is more efficient in practice.

Revocation is a challenge problem in ABE-based access control system [18]. Directly revocation is applied in [1], while in [4] an indirectly revocation is proposed. Effectively revocation is added in KP-ABE scheme in the indirect mode [12].Cui and Deng used the indirectly revocation technique into a decentralized CP-ABE scheme to realize fine-grained revocation [8]. In NDN, the ciphertext updating caused by directly revocation cannot be completed synchronously in each cache of NDN routers. Therefore, the indirect revocation mechanism is more suitable for NDN.

The first CP-ABE scheme with anonymity proposed in 2008 [21], where the access policy only supports AND gates structure. Recently, a CP-ABE scheme with anonymous access structure is proposed [13], which can be expressed as any LSSS [20] matrix and fully security.

Based on Rouselakis's work [23], combining with in-network storage strategy in NDN, this paper proposes a

Table 1: Summary of the access control schemes in NDN

Schemes	Based on	Revocation	Always-online Publisher
Chen *et al.*[5]	Encryption	Re-encryption	Need
Wood *et al.* [29]	Proxy re-encryption	Not consider	Need
Tan *et al.*[7]	Encryption	Considered	Need
Hamdane *et al.* [11]	Encryption & Credential	Not consider	Need
Ghali *et al.* [9]	Interest-based	Not consider	Need
Li *et al.* [17]	Signature-based	Not consider	Not need
Silva *et al.* [25]	Attribute encryption & Proxy re-encryption	Proxy re-encryption	Proxy is Needed

new access control system for NDN using decentralized CP-ABE, which is revocable and privacy-preserving.

3 Preliminaries

3.1 Access Structure

Definition 1 (Access Control [13]). *Let $\{P_1, \ldots, P_n\}$ be a set of all parties. A collection $\mathbb{A} \subseteq 2^{\{P_1,\ldots,P_n\}}$ is monotone if $\forall B, C$: if $B \in \mathbb{A}$ and $B \subseteq C$, then $C \in \mathbb{A}$. An access structure is a collection $\mathbb{A}$ of non-empty subsets of $\{P_1, \ldots, P_n\}$, i.e., $\mathbb{A} \subseteq 2^{\{P_1,\ldots,P_n\}} \backslash \{\varnothing\}$. Authorized sets are the sets in $\mathbb{A}$, and unauthorized sets are the sets not in $\mathbb{A}$.*

In this work, the parties represent attributes in our scheme. And we only consider monotone access structure, which means that consumers will not lose their previous decryption privileges if they get more attributes.

3.2 Linear Secret Sharing Scheme (LSSS) [20]

Definition 2 (Linear secret sharing scheme (LSSS) [13]). *Let $\mathcal{U}$ denote a set of parties, $\mathcal{U}$ is the attribute universes in our context. Let $\boldsymbol{A}$ be a matrix of size $l \times n$. Let $\rho : \{1, \ldots, l\} \to \mathcal{U}$ be a function that maps a row to a party for labeling. A secret sharing scheme $\prod$ over a set of parties $\mathcal{U}$ is a linear secret-sharing scheme over $\mathbb{Z}_p$ (p is a prime) if:*

1) *The shares of a secret $s \in \mathbb{Z}_p$ for each attribute form a vector over $\mathbb{Z}_p$.*

2) *For each access structure $\mathbb{A}$ on $\mathcal{U}$, there exists a matrix $\mathbf{A} \in \mathbb{Z}_p^{l \times n}$, called the share-generating matrix for $\prod$. For all $i = 1, \ldots, l$, the function ρ labels the rows of $\boldsymbol{A}$ with attributes from $\mathcal{U}$, i.e., the i^{th} row of $\boldsymbol{A}$ is labeled by $\rho(i)$. When we consider the column vector $\vec{v} = (s, r_2, \ldots, r_n)$, where $s \in \mathbb{Z}_p$ is the secret to be shared, and $r_2, \ldots, r_n \in \mathbb{Z}_P$ are randomly chosen, then $\boldsymbol{\lambda} = \boldsymbol{A}\vec{v} \in \mathbb{Z}_p^{l \times 1}$ is the vector of l shares of the secret s according to $\prod$. The share $\boldsymbol{l}ambda_i = \boldsymbol{A}_i\vec{v}$ belongs to party $\rho(i)$.*

As addressed in [13], each secret-sharing scheme (not only the linear ones) should satisfy the reconstruction requirement (each authorized set can reconstruct the secret) and the security requirement (any unauthorized set cannot reveal any partial information about the secret). More concretely, let $S \in \mathbb{A}$ be an authorized set, let I be the set of rows whose labels are in S and define $I = \{i | \rho(i) \in S\} \subset \{1, \ldots, l\}$. Then there exist constants $\{c_i \in \mathbb{Z}_p\}$ such that, if $\{\lambda_i\}$ are valid shares of any secret s according to $\prod$, then $\sum_{i \in I} c_i \lambda_i = s$, i.e., $\sum_{i \in I} c_i \boldsymbol{A}_i = (1, 0, \ldots, 0)$. These constants $\{\lambda_i\}$ can be found in time polynomial in the size of share-generation matrix $\boldsymbol{A}$. But for unauthorized sets, no such $\{\lambda_i\}$ constants exist.

3.3 Bilinear Groups and Complexity Assumption

Our scheme is constructed on bilinear groups of prime order, and its security is proved based on q-Decisional Parallel Bilinear Diffie-Hellman Exponent 2 (q-BPBDHE2).

Let G and G_T be two multiplicative cyclic groups of prime order p, where the group operation is efficiently computable in the security parameter. Let g be a generator of G and $e : G \times G \to G_T$ be a bilinear map that satisfies the following properties:

1) **Bilinearity**: for all $g, f \in G$ and $a, b \in \mathbb{Z}_p$ it is true that $e(g^a, f^b) = e(g, f)^{ab}$.

2) **Non-degeneracy**: $e(g, g) \neq 1_{G_T}$.

3) **Computability:** The group operations in G and the bilinear map $e : G \times G \to G_T$ are both efficiently computable.

Definition 3 (q-BPBDHE2 [23]). *G is a bilinear group of order p and g is a generator of G. The q-BPBDHE2 problem in group G is defined as follows: Choose random values $s, a, b_1, b_2, \ldots, b_q \in Z_p^*$ and $R \in G_T$, and given*

$$D = ((p, g, G, e, g^s, \{g^{a^i}\}_{i \in [2q], i \neq q+1}, \{g^{b_j a^i}\}_{(i,j) \in [2q,q], i \neq q+1},$$
$$\{g^{\frac{s}{b_i}}\}_{i \in [q]}, \{g^{\frac{sa^i b_j}{b_{j'}}}\}_{(i,j,j') \in [q+1,q,q], j \neq j'})$$

The assumption states in G that no polynomial-time distinguisher can distinguish the distribution $(D, e(g,g)^{sa^{q+1}})$ from the distribution (D, R) with more than negligible advantage.

4 Access Control System in NDN

4.1 Access Control System Model

This paper considers a NDN environment consisting of AAs, data publisher, NDN routers and data consumers. The system model of this NDN access control system is showed in Figure 2.

1) **Attribute Authorities (AAs):** Each AA creates a pair of public-private key in setup phase using public parameters, and excepting this, no other global coordination is needed. Each AA is independent from other AAs and manages different attributes. In this scheme, each AA can control any number of attributes, while each attribute can be managed only by one AA. While receiving a request for the attribute-related key from a consumer, AA responds the key under current time period. Additionally, at the beginning of each time period, they update the attribute-related keys for the non-revoked consumers based on a revocation list.

2) **Data publisher:** A data publisher can develop access policy by himself, and then encrypts data according to this policy and the current time period, and specifies how long (i.e., the current time period) the content can be cached in routers through "freshness period" in data packet. The attribute policy is sent with ciphertext but the attribute values are always hidden.

3) **NDN routers:** They can provide data storage service by its own storage strategy, and give consumers data packets according to their interest packets. Moreover, they can also delete the data packets according to the "freshness period" in data packets.

4) **Data consumers:** Each consumer with a GID is represented by an attribute set. Consumers can ask the AAs for attribute-related keys under the current time period. A consumer can obtain the encrypted data from his neighbor router or the publisher, and then according to the access structure in ciphertext, if he has an attributes set matching the access structure and all the attribute-related keys are under the right time period can decrypt the ciphertext. No one can learn about the specific access structure.

4.2 Definition of the Access Control Scheme

This scheme includes the following five parts: GlobalSetup, AASetup, KeyGen & KeyUpdate, Encrpt and Decrpt. Revocation is an important part in access control system. In the ABE-based access control system, revocation occurs when a consumer leaves from the system by himself, a leaker was deleted by system, a consumer's attribute expired, or an attribute in the system was abolished. In our scheme, the attribute-based keys of the non-revoked consumers should be updated periodically, and the period is static and determined by the system requirements, for example updated every month. The publisher encrypts the data under his own policy and the current time period. Each attribute is comprised of two parts, name and value. In addition, a revocation list for each attribute value is needed in the system, which stores the GIDs of all the revoked consumers associated with the attribute value, and is stored by the corresponding AA.

GlobalSetup$(1^\kappa) \rightarrow GP$: κ is the security parameter, GP is the output representing public global parameters. The attribute name universe $\mathcal{U}$, AA universe $\mathcal{U}_\theta$, global identifier universe $\mathcal{GID}$ and a publicly computable function F ($F : \mathcal{U} \rightarrow \mathcal{U}_\theta$ map each attribute name to a unique authority) are all included in GP and $\theta \in \mathcal{U}_\theta$.

AASetup$(GP, \theta) \rightarrow \{apk_\theta, ask_\theta\}$: $AA(\theta \in \mathcal{U}_\theta)$ runs it which takes in GP and generates its public-secret key pair (apk_θ, ask_θ).

KeyGen & $KeyUpdate(GP,\ GID,\ u,\ v_u,\ Time,\ rl_{v_u},\ ask_\theta) \rightarrow \{SK^{Time}_{GID,v_u}\}$: The consumer asks for an attribute-related key from the corresponding authority θ, and θ runs this algorithm which takes in the consumer's global identifier $GID(GID \in \mathcal{GID})$, a name u of attribute and its value v_u , the current time period Time, a revocation list rl_{v_u} related to v_u and ask_θ of θ. It generates a key SK^{Time}_{GID,v_u} for this consumer.

Encrpt$(GP, \mathrm{M}, (\boldsymbol{A}, \rho, \mathcal{T}), Time, \{apk_\theta\}) \rightarrow$ CT: Publisher runs this algorithm, which takes in the content M, the current time period $Time$, the access structure developed by the publisher $(\boldsymbol{A}, \rho, \mathcal{T})$ and a public key set $\{apk_\theta\}$of the corresponding authorities. It outputs the ciphertexts CT, and the part of access structure $(\boldsymbol{A}, \rho)$ are passed with CT, but the attribute values $\mathcal{T}$ are always hidden.

Decrpt$(GP, \mathrm{CT}, \{SK^{Time}_{GID,v_u}\}) \rightarrow$ M: when a consumer obtains an encrypted data, he runs this algorithm, which takes in CT and an attribute-related key set $\{SK^{Time}_{GID,v_u}\}$ of the consumer GID corresponding to different attributes value. The consumer checks which attributes are used to match the access structure, and then employs these keys to decrypt the ciphertexts, but never know the specific access structure.

4.3 Security Requirements

1) Effective access control:
Publishers publish their sensitive content to NDN and have an expressive access policy for published content, so that the specific consumers can decrypt it. Each consumer can obtain the encrypted data packet from the neighbor router or the publisher, but

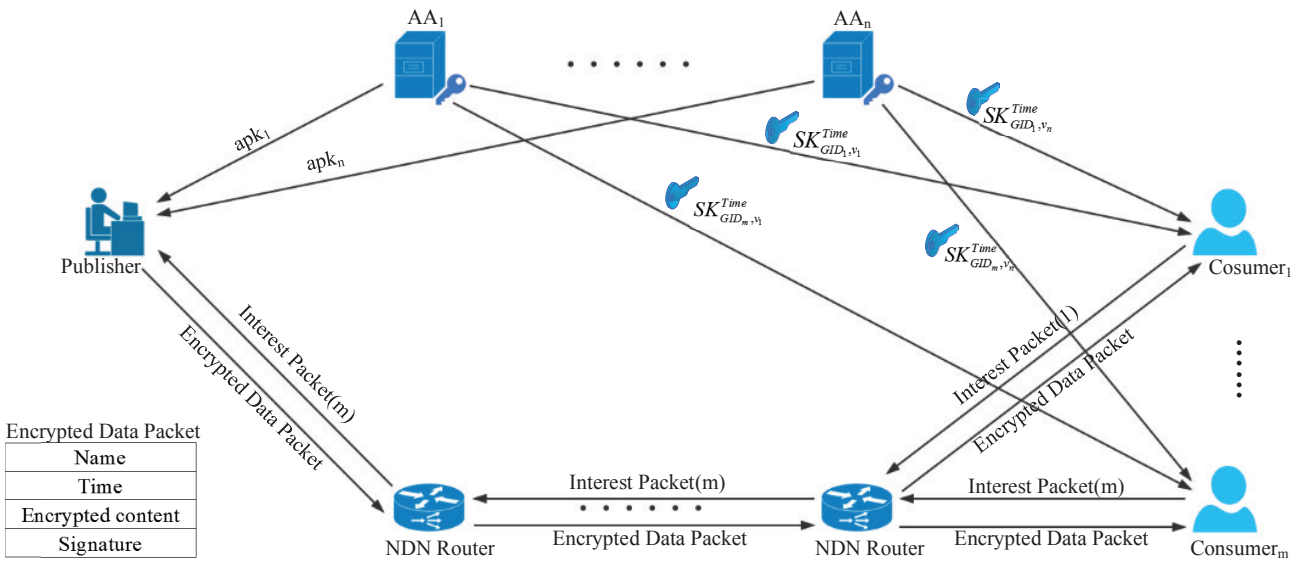


Figure 2: Architecture of our NDN access control system

only the authorized consumers can decrypt it.

2) Collusion resistance:
Sometimes, although an unauthorized consumer cannot decrypt the ciphertext alone, several of them may decrypt it by collaborating with each other, which is called collusion attack. This access system is required to against this sort of attack.

3) Back security and forward security:
This access control system is required that both backward security and forward security are guaranteed. That is, if the private key is compromised, the ciphertext encrypted in the last time period and next period are not affected.

4) Privacy-preserving:
In many applications, some specific attributes carry sensitive information, but the access structure included in ciphertext is public. So, it is required that the access structure should not reveal consumer's sensitive information to realize user privacy.

4.4 Security Model

This scheme is improved the CP-ABE scheme in [23] basically, and its security is proved based on q-Decisional Parallel Bilinear Diffie-Hellman Exponent 2 (q-BPBDHE2) [23], and a static (or non-adaptive) security model is also used. As defined in [23], in the static security model, the adversary will send all queries to the challenger the first time after knowing the public parameters. Furthermore, the adversary is allowed to destroy a set of AAs for the purpose of malicious attacks, and the adversary selects these AAs after knowing the public parameters and remain unchanged during the course of the game. The formalized security game is specified below:

Global Setup: Challenger runs GlobalSetup$(1^\kappa) \rightarrow GP$ and gives GP to adversary.

Adversary's Queries: Then the adversary responds with:

- A set $\mathcal{C}_\theta \subseteq \mathcal{U}_\theta$ of corrupt AAs chosen by the adversary and their respective public keys $\{apk_\theta\}_{\theta \in C_\theta}$ which can be created in a malicious way but in correct type.
- A set $\mathcal{N}_\theta \subseteq \mathcal{U}_\theta$ of the non-corrupt AAs which is disjoint from $\mathcal{C}_\theta \subseteq \mathcal{U}_\theta$. And the adversary queries their public keys.
- A sequence $\{(\mathcal{S}_i, GID_i), Time\}_{i=1}^m$ for querying the secret keys, in which GID_i is a consumer's global identity and $\mathcal{S}_i \subseteq \mathcal{U}$ is an attribute set. The adversary queries a pair $\{(\mathcal{S}_i, GID_i), Time\}$ means that he asks for the secret keys for the consumer GID_i under the current time period $Time$, and the consumer's attribute set is $\mathcal{S}_i$. That is, the identities $\{GID_i\}$ are distinct and all the keys are come from a non-corrupt AA, i.e.,$F(\mathcal{S}_i) \cap \mathcal{C}_\theta = \varnothing$.
- Two messages $\mathrm{M}_0, \mathrm{M}_1$ with equal length and a challenge access structure $(\boldsymbol{A}, \rho, \mathcal{T})$ ($\boldsymbol{A}$ is a LSSS matrix of access structure, ρ is a function mapping the rows in $\boldsymbol{A}$ to attribute names). It is required for each GID_i, the access structure $(\boldsymbol{A}, \rho, \mathcal{T})$ cannot be satisfiede by $\mathcal{S}_i \cup \mathcal{S}_{\mathcal{C}_\theta}$, where $\mathcal{S}_{\mathcal{C}_\theta}$is a collection of all the attributes managed by corrupt AAs. So, the adversary cannot decrypt the challenge ciphertext by combining a secret key given to him with the keys from the corrupt AAs to win the game.

Challenger's Replies: The challenger throws a random coin $\beta \in \{0, 1\}$ and replies with:

- The public keys $\{apk_\theta\}_{\theta\in\mathbb{N}_\theta}$ corresponding to the non-corrupt authorities $\mathcal{N}_\theta$.
- The secret keys $\{(SK^{Time}_{GID_i,\mathcal{S}_i})\}^m_{i=1}$ corresponding to $\{(\mathcal{S}_i, GID_i), Time\}^m_{i=1}$.
- The challenge ciphertext Encrypt(GP, M_β, $\{apk_\theta\}$ $\{\boldsymbol{A}, \rho, \mathcal{T}\}$) $\rightarrow$ CT$*$, in which $\{apk_\theta\}$ is the set of all the corresponding AA's public keys.

Guess: The adversary guesses β' for β as output.

The advantage of the adversary in the game is defined as $\Pr[\beta=\beta']-1/2$, that is the probability of the adversary guessing correctly.

Definition 4. *The scheme is statically secure if no polynomial time adversary has a non-negligible advantage to win the above security game.*

5 Construction

The detailed algorithms of this scheme are constructed as follows,

GlobalSetup(1^κ) $\rightarrow GP$: It takes the security parameter κ as input. First, it chooses a bilinear group G which order is a prime p and generator is g. Three Hash functions are used in this algorithm, $H_0 : \{0,1\}^* \rightarrow \mathbb{Z}_p$, $H_1 : \mathbb{Z}^*_P \rightarrow G$, $H_2 : \mathcal{U} \rightarrow G$, H_0 maps time period to elements of $\mathbb{Z}_p$, H_1 and H_2 maps global identities GID and attribute names to G respectively. Finally, it defines $F : \mathcal{U} \rightarrow \mathcal{U}_\theta$. GP= $\{p, G, g, q, H_0, H_1, H_2, \mathcal{U}, \mathcal{U}_\theta, F\}$ are global parameters as output. If not specified, GP is given as an input in the following algorithms.

AASetup(GP, θ) $\rightarrow \{apk_\theta, ask_\theta\}$**:** In this algorithm, each AA ($\theta \in \mathcal{U}_\theta$) chooses two random exponents $\alpha_\theta, y_\theta \in \mathbb{Z}_p$. It keeps $ask_\theta = \{\alpha_\theta, y_\theta\}$ as its secret key. The AA publishes its pulic key $apk_\theta = \{e(g,g)^{\alpha_\theta}, g^{y_\theta}\}$.

KeyGen&KeyUpdate(GP**,** GID**,** θ**,** u**,** v_u**,** $Time$**,** rl_{v_u}**,** ask_θ) $\rightarrow \{SK^{Time}_{GID,v_u}\}$**:** AA runs this algorithm and takes in the consumer's global identifier GID, the attribute name u and the corresponding attribute value v_u, the current time period Time, the revocation list related to v_u and the authority's secret key to create or update a key for the consumer. The attribute is managed by the specific authority, i.e., $u \in F^{-1}(\theta)$. It first chooses a $\gamma \in \mathbb{Z}_p$ randomly and generates the attribute-related key under the current time period as output: $SK^{Time}_{GID,v_u} = \{K = g^{\alpha_\theta H_0(Time)} H_1(GID \parallel Time)^{y_\theta} H_2(u)^\gamma, K' = g^{v_u\gamma}\}$.

Encrpt(GP, M, $(\boldsymbol{A}, \rho, \mathcal{T})$, $Time$, $\{apk_\theta\}$) $\rightarrow$ CT :
Publisher runs this algorithm, takes in a content M with sensitive information, an access policy $(\boldsymbol{A}, \rho, \mathcal{T})$ with $\boldsymbol{A} \in \mathbb{Z}_p^{l\times n}$, where ρ is a function mapping the rows in $\boldsymbol{A}$ to the attribute names and $\mathcal{T} = (t_{\rho(1)}, \cdots, t_{\rho(l)}) \in \mathbb{Z}^l_p$ is the corresponding attribute values, the current time period $Time$ and the public key sets $\{apk_\theta\}$ of the related AAs. Also, the function is defined as $\delta : [l] \rightarrow \mathcal{U}_\theta$ as $\delta(\cdot) = F(\rho(\cdot))$, that is, mapping the rows of $\boldsymbol{A}$ to AAs.

It chooses vector $\vec{v} = (s, v_2, \cdots, v_n)^T$, $\vec{v}' = (s', v'_2, \cdots, v'_n)^T$ and $\vec{\omega} = (0, \omega_2, \cdots, \omega_n)^T$, $\vec{\omega}' = (0, \omega'_2, \cdots, \omega'_n)^T$ at first, where s, $v_2, \cdots, v_n$, $s', v'_2, \cdots, v'_n \in \mathbb{Z}_p$ and $\omega_2, \cdots, \omega_n$, $\omega'_2, \cdots, \omega'_n \in \mathbb{Z}_p$ are chosen randomly. According to LSSS, let λ_x denote the share of s, i.e. $\lambda_x = \boldsymbol{A}_x\vec{v}$, and ω_x denote the share of 0, i.e. $\omega_x = \boldsymbol{A}_x\vec{\omega}$, where $\boldsymbol{A}_x$ is the x-th row of $\boldsymbol{A}$.

For each $x (x \in [l])$ of $\boldsymbol{A}$, choose $r_x, r'_x \in \mathbb{Z}_p$ randomly. The ciphertext is computed as:

$$
\begin{aligned}
C_0 =& Me(g,g)^s \quad C'_0 = e(g,g)^s \\
\forall x \quad & C_{1,x} = e(g,g)^{\lambda_x} e(g,g)^{\alpha_{\rho(x)} H_0(Time) r_x t_{\rho(x)}} \\
& C_{2,x} = g^{-r_x t_{\rho(x)}} \\
& C_{3,x} = g^{y_{\rho(x)} r_x t_{\rho(x)}} g^{\omega_x} \\
& C_{4,x} = H_2(\rho(x))^{r_x} \\
& C'_{1,x} = e(g,g)^{\lambda'_x} e(g,g)^{\alpha_{\rho(x)} H_0(Time) r'_x t_{\rho(x)}} \\
& C'_{2,x} = g^{-r'_x t_{\rho(x)}} \\
& C'_{3,x} = g^{y_{\rho(x)} r'_x t_{\rho(x)}} g^{\omega'_x} \\
& C'_{4,x} = H_2(\rho(x))^{r'_x}
\end{aligned}
$$

It outputs ciphertext CT as:

$$
\begin{aligned}
\mathrm{CT} = ((\boldsymbol{A}, \rho), C_0, C'_0, \{C_{1,x}\}, \{C'_{1,x}\}, \{C_{2,x}\}, \{C'_{2,x}\}, \\
\{C_{3,x}\}, \{C'_{3,x}\}, \{C_{4,x}\}, \{C'_{4,x}\})
\end{aligned}
$$

Finally, the publisher signs the data packet and publish it or waits for the consumer's request.

Decrpt(GP, CT, $\{SK^{Time}_{GID,v_u}\}$) $\rightarrow$ M : When a consumer obtains the encrypted data packet, after verifying the signature of publisher to ensure the integrity and authenticity of the content. Then, the consumer runs this algorithm calculates $\min_{(\boldsymbol{A},\rho)}$ from $(\boldsymbol{A}, \rho)$, in which $\min_{(\boldsymbol{A},\rho)}$ is the minimum subsets matches $(\boldsymbol{A}, \rho)$. It then checks if there is a $\mathcal{L} \in \min_{(A,\rho)}$ that satisfies

$$
\begin{aligned}
C'_0 = \prod_{x\in L} (C'_{1,x} \cdot e(K, C'_{2,x}) \cdot e(H_1(GID \parallel Time), C'_{3,x}) \cdot \\
e(K', C'_{4,x}))^{c_x}
\end{aligned}
$$

Where $\sum_{x\in L} c_x \boldsymbol{A}_x = (1, 0, \cdots, 0)$. If no element in $\min_{(\boldsymbol{A},\rho)}$ makes the above equation holds, which means the consumer is unauthorized, then outputs $\perp$, that is, it is failed to decrypt. Otherwise, the consumer calculates constants $c_x \in \mathbb{Z}_p$ such that

$\sum_{x\in I} c_x \boldsymbol{A}_x = (1,0,\ldots,0)$ and computes:

$$\prod_x (C_{1,x}\cdot e(K,C_{2,x})\cdot e(H_1(GID \parallel Time), C_{3,x})\cdot e(K', C_{4,x}))^{c_x} = e(g,g)^s$$

Since $\lambda_x = \boldsymbol{A}_x\vec{v}$ and $\omega_x = \boldsymbol{A}_x\vec{\omega}$, such that $(1,0,\ldots,0)\cdot\vec{v} = s$ and $(1,0,\ldots,0)\cdot\vec{\omega} = 0$. Then M can be recovered as $\mathrm{M} = C_0/e(g,g)^s$.

In the construction, each available AA updates the consumer's attribute-related key periodically. It means that all consumers should contact AAs to get new private keys periodically. This can be extremely complex if the number of users is large. To avoid establishing the secure channel, let each AA encrypt the new key for the non-revoked consumer with its GID and the previous time period, and pass the encrypted key to the consumer.

6 Security Analysis

6.1 Correctness

Theorem 1. *This scheme is correct.*

Proof. If there is a $\mathcal{L} \in \min_{(A,\rho)}$,

$$\begin{aligned}
& C'_{1,x}\cdot e(K,C'_{2,x})\cdot e(H_1(GID \parallel Time), C'_{3,x}) \cdot e(K', C'_{4,x}) \\
= \; & e(g,g)^{\lambda'_x}\cdot e(g,g)^{\alpha_{\rho(x)}H_0(Time)r'_x t_{\rho(x)}}\cdot \\
& e(g^{\alpha_{\rho(x)}H_0(Time)} H_1(GID \parallel Time)^{y_{\rho(x)}} H_2(\rho(x))^{\gamma}, \\
& g^{-r'_x t_{\rho(x)}})\cdot e(H_1(GID \parallel Time), g^{y_{\rho(x)}r'_x t_{\rho(x)}} g^{\omega'_x}) \\
& \cdot e(g^{\gamma t_{\rho(x)}}, H_2(\rho(x))^{r'_x}) \\
= \; & e(g,g)^{\lambda'_x}\cdot e(g,g)^{\alpha_{\rho(x)}H_0(Time)r'_x t_{\rho(x)}}\cdot \\
& e(g,g)^{-\alpha_{\rho(x)}H_0(Time)r'_x t_{\rho(x)}}\cdot \\
& e(H_1(GID \parallel Time), g)^{-y_{\rho(x)}r'_x t_{\rho(x)}}\cdot \\
& e(H_2(\rho(x)), g)^{-\gamma r'_x t_{\rho(x)}} \\
& \cdot e(H_1(GID \parallel Time), g)^{y_{\rho(x)}r'_x t_{\rho(x)}}\cdot \\
& e(H_1(GID \parallel Time), g)^{\omega'_x}\cdot e(g, H_2(\rho(x)))^{\gamma r'_x t_{\rho(x)}} \\
= \; & e(g,g)^{\lambda'_x}\cdot e(H_1(GID \parallel Time), g)^{\omega'_x}
\end{aligned}$$

Then calculates $c_x \in \mathbb{Z}_p$ such that $\sum_{x\in L} c_x \boldsymbol{A}_x = (1,0,\ldots,0)$.

$$\sum_{x\in L}\lambda'_x c_x = \sum_{i\in I}\mathbb{A}_i\cdot\vec{v}'\cdot c_x = \vec{v}'\cdot(1,0,\ldots,0) = s',$$
$$\sum_{x\in L}\omega'_x c_x = \sum_{i\in I}\mathbb{A}_i\cdot\vec{\omega}'\cdot c_x = \vec{\omega}'\cdot(1,0,\ldots,0) = 0.$$

So,

$$\begin{aligned}
& \prod_{x\in\mathcal{L}} (e(g,g)^{\lambda'_x} e(H_1(GID \parallel Time), g)^{\omega'_x})^{c_x} \\
= \; & e(g,g)^{\sum_{x\in I}\lambda'_x c_x} e(H_1(GID \parallel Time), g)^{\sum_{x\in I}\omega'_x c_x} \\
= \; & e(g,g)^{s'} = C'_0
\end{aligned}$$

Similarly, calculates

$$\prod_{x\in\mathcal{L}} (C_{1,x}\cdot e(K,C_{2,x})\cdot e(H_1(GID \parallel Time), C_{3,x})\cdot e(K', C_{4,x}))^{c_x} = e(g,g)^s$$

Finally, M can be recovered as

$$\begin{aligned}
& C_0/\prod_{x\in\mathcal{L}} (e(g,g)^{\lambda_x} e(H_1(GID \parallel Time), g)^{\omega_x})^{c_x} \\
= \; & C_0/e(g,g)^s = \mathrm{M}
\end{aligned}$$

□

6.2 Static Security

In this scheme, publisher can develop his own access policy at will for the sensitive information. Each AA can publish attribute-related keys for the consumers. Each consumer can obtain the published data packet from neighbor router conveniently without affecting in-network storage mechanism, and only the authorized consumers can decrypt the content, while unauthorized consumers can get none of useful information from the acquired data packet.

The static security of this scheme will be proved from q-DPDDHE2 assumption under the static security model. First, the following lemma will be proved.

Lemma 1. *Supposing that the scheme in [23] is a statically secure scheme, then this scheme is a static secure one.*

Proof. Assuming there is a polynomial-time $\mathscr{A}$ who has advantage ε against this scheme in the static security game. Then how to build a simulator $\mathscr{B}$ that has advantage ϵ against the scheme in [23] is as follows. Let $\mathscr{C}$ be the challenger in [23]. □

GlobalSetup: The challenge $\mathscr{C}$ sends the global parameters $GP = \{p, G, g, H_0, H_1, H_2, \mathcal{U}, \mathcal{U}_\theta, F\}$ to the simulator $\mathscr{B}$. $\mathscr{B}$ passes GP to the adversary $\mathscr{A}$.

Adversary's Queries: The adversary $\mathscr{A}$ chooses a set $\mathcal{C}_\theta \subseteq \mathcal{U}_\theta$ of corrupt AAs and generates the related public keys in the scheme [23] as $\{apk'_\theta\}_{\theta\in\mathcal{C}_\theta}$. For each $\theta \in \mathcal{C}_\theta$, A sets the keys of corrupt AAs in our scheme as $apk_\theta = \{apk'_\theta\}$. $\mathscr{A}$ responds to $\mathscr{B}$ with:

- A set of corrupt AAs $\mathcal{C}_\theta \subseteq \mathcal{U}_\theta$ and $\{apk_\theta\}_{\theta\in\mathcal{C}_\theta}$.
- A non-corrupt AAs set $\mathcal{N}_\theta \subseteq \mathcal{U}_\theta$.
- A sequence $\{(S_i, GID_i), Time\}_{i=1}^m$ with the following restrictions: if $i \neq j$, then $GID_i \neq GID_j$, $S_i \subseteq \mathcal{U}$ and $F(S_i) \cap \mathcal{C}_\theta = \varnothing$. A pair $\{(S_i, GID_i), Time\}$ means that $\mathscr{A}$ requests the secret key for the consumer with global identity GID_i under the current time period Time, the consumer's attributes set is S_i.

- Two messages $\mathrm{M}_0, \mathrm{M}_1$ with equal length, and a challenge access structure $(\boldsymbol{A}, \rho)$. $\mathcal{S}_{\mathcal{C}_\theta}$ is a collection of all the attributes managed by corrupt AAs. For each $i \in [m]$, the attribute in S_i cannot be combined with the attribute in $S_{\mathcal{C}_\theta}$ (i.e., $S_{\mathcal{C}_\theta} \cup S_i$) to satisfy $(\boldsymbol{A}, \rho)$.

Challenger's Replies: When the simulator $\mathscr{B}$ receives the above responds, it sends $\mathcal{C}_\theta$, $\{apk_\theta\}_{\theta \in \mathcal{C}_\theta}$, $\mathcal{N}_\theta$, $\{(S_i, GID_i), Time\}_{i=1}^m$, $\mathrm{M}_0, \mathrm{M}_1$ and $(\boldsymbol{A}, \rho)$ to $\mathscr{C}$ for requesting the corresponding public keys, secret keys and challenge ciphertext in the scheme in [23]. Then, $\mathscr{C}$ replies the public keys $\{apk'_\theta = (e(g,g)^{\alpha_\theta}, g^{y_\theta})\}$ for all $\theta \in \mathcal{N}_\theta$, the secret keys $SK'_{GID_i,S_i} = \{(g^{\alpha_\theta} H_1(GID_i)^{y_\theta} F(i)^t, g^t)_{i \in s_i}\}$, for all $i \in [m]$, the challenge ciphertext $\mathrm{CT}' = (C_0 = \mathrm{M}_\beta e(g,g)^s, \{C_{1,x} = e(g,g)^{\lambda_x} e(g,g)^{\alpha_{\rho(x)} r_x}\}, \{C_{2,x} = g^{-r_x}\}, \{C_{3,x} = g^{y_{\rho(x)} r_x} g^{\omega_x}\}, \{C_{4,x} = F(\rho(x)^{t_x}\}_{x \in \{1,2,\ldots,l\}})$. Then $\mathscr{B}$ generates the public keys, the secret keys and challenge ciphertext in our scheme as follows:

- For each $\theta \in \mathcal{N}_\theta$, $\mathscr{B}$ sets the public keys as $\{apk_\theta = (e(g,g)^{\alpha_\theta}, g^{y_\theta})\}$.
- For each $i \in [m]$ and $u \in \mathcal{S}_i$, a random value t is chosen randomly by $\mathscr{B}$, the current time period $Time$, then calculates $K = g^{\alpha_\theta H_0(Time)} H_1(GID \parallel Time)^{y_\theta} H_2(u)^\gamma$, $K' = g^{v_u \gamma}$. Finally, $\mathscr{B}$ sets the attribute-related keys as $SK^{Time}_{GID,v_u} = \{K = g^{\alpha_\theta H_0(Time)} H_1(GID \parallel Time)^{y_\theta} H_2(u)^\gamma, K' = g^{v_u \gamma}\}$.
- For $x \in \{1, 2, \ldots, l\}$, $\mathscr{B}$ calculates $C_0 = Me(g,g)^s$, $C'_0 = e(g,g)^s$, $C_{1,x} = e(g,g)^{\lambda_x} e(g,g)^{\alpha_{\rho(x)} H_0(Time) r_x t_{\rho(x)}}$, $C_{2,x} = g^{-r_x t_{\rho(x)}}$, $C_{3,x} = g^{y_{\rho(x)} r_x t_{\rho(x)}} g^{\omega_x}$, $C_{4,x} = H_2(\rho(x))^{r_x}$, $C'_{1,x} = e(g,g)^{\lambda'_x} e(g,g)^{\alpha_{\rho(x)} H_0(Time) r'_x t_{\rho(x)}}$, $C'_{2,x} = g^{-r'_x t_{\rho(x)}}$, $C'_{3,x} = g^{y_{\rho(x)} r'_x t_{\rho(x)}} g^{\omega'_x}$, $C'_{4,x} = H_2(\rho(x))^{r'_x}$. B sets the challenge ciphertext as

$$\begin{aligned}\mathrm{CT} = &((\boldsymbol{A}, \rho), C_0, C'_0, \{C_{1,x}\}, \{C'_{1,x}\}, \{C_{2,x}\},\\ &\{C'_{2,x}\}, \{C_{3,x}\}, \{C'_{3,x}\}, \{C_{4,x}\}, \{C'_{4,x}\}).\end{aligned}$$

Finally, $\mathscr{B}$ sends $\{apk_\theta = (e(g,g)^{\alpha_\theta}, g^{y_\theta})\}$, $\{SK^{Time}_{GID_i,S_i}\}_{i=1}^m$ and the challenge ciphertexts CT to $\mathscr{A}$.

Guess: $\mathscr{A}$ guesses $\beta' \in \{0,1\}$ as output. Then $\mathscr{B}$ outputs β'.

And in [23], the following lemma has been proved.

Lemma 2. *If the q-DPBDHE2 holds, the scheme in [23] is statically security in random oracle model.*

Theorem 2. *If the q-DPBDHE2 holds, then this scheme is statically security in the random oracle model.*

Proof. It can be proved directly from **Lemma 1** and **Lemma 2.** □

6.3 Collusion Resistance

Theorem 3. *This scheme can against the collusion attack by the unauthorized consumers.*

Proof. It can be against the collusion attacks by combining a consumer's key components together by distinct global identity GID. Furthmore, when encrypting a message, besides specifing a access stucture, the data publisher also needs to specify the current time period, and a consumer whose attribute-related key satisfies the structure and under the current time period can decrypt the ciphertext. Since the time is same for each consumer, so that the revoked consumers may decrypt the newly created ciphertext by combining his key with the updated information of non-revoked consumers, so it is necessary to prevent the consumers from collusion. Therefore, $SK^{Time}_{GID,v_u} = \{K = g^{\alpha_\theta H_0(Time)} H_1(GID \parallel Time)^{y_\theta} H_2(u)^\gamma, K' = g^{v_u \gamma}\}$ are generated as the attribute-related key.The time period and a consumer's global identity are bound together as $H_1(GID \parallel Time)$ which is different from each other but related to his or her own identity. □

6.4 Back Security and Forward Security

Theorem 4. *This scheme can guarantee back security and forward security.*

Proof. The attribute-related keys and ciphertexts are updated periodically, and the value of parameter $Time$ is different for each time period. Therefore, the ciphertext encrypted in previous time period cannot be decrypted with the updated private keys of the current time period, which ensures the backward security. In this time period, the re-encrypted ciphertext in this time period cannot be decrypted with the attribute-related keys distributed in previous time period, which guarantees the forward security. □

6.5 Privacy-preserving

Theorem 5. *The scheme can realize recipient anonymity to protect privacy of users.*

Proof. In the access policy $(\boldsymbol{A}, \rho, \mathcal{T})$, $\boldsymbol{A}$ is the access matrix, ρ is a function mapping the rows in $\boldsymbol{A}$ to attribute names, $\mathcal{T}$ is the concrete attribute value. In our scheme, the attribute value $\mathcal{T}$ is always hidden, but the rest of the access structure $(\boldsymbol{A}, \rho)$ is sends with the ciphertext. No one can learn the specific access structure in the ciphertext, so even legitimate consumers don't know which attributes are used to decrypt the data. Therefore, no one knows who can decrypt the ciphertext to realize recipient anonymity, and thus it protects the user privacy. □

7 Characteristics Comparison

The scheme in [23] is a large-universe multi-authority CP-ABE scheme based on prime order groups which re-

duces the computational complexity observably. Based on the scheme in [23], this paper combine the indirect revocation method and the NDN architecture to realize revocation in NDN. Meanwhile, this scheme is preserving-privacy through using the partially hidden structure technique. As shown in Table 2, a comparison of characteristics of this paper with those in [8, 13, 23] is described. [23] is the basic scheme we implement and improvement, the scheme in [8] is a multi-authority CP-ABE scheme with indirectly revocation, and the scheme in [13] is an anonymity one. From Table 2, we can observe our scheme is fully functional, it is a decentralized CP-ABE scheme supporting both revocation and anonymity. Also the prime order groups brings it efficient and practical features.

We use $|\mathcal{U}|$ denote the size of attribute universe. The number of attribute authorities is denoted by $|\mathcal{U}_\theta|$, an LSSS access structure with an $l \times n$ matrix is represented by l, the number of attributes in the consumer's key is denoted by $|\mathcal{S}|$, and the number of rows used in decryption is denoted by $|I|$. Table 3 summarizes the efficiency of our scheme and the other schemes.

In contrast to the basic scheme [23], the public and private key size in this paper are not changed, but since the need of redundant ciphertexts for hidden structure, the ciphertext length and bilinear operation for decryption are twice as much as the basic scheme. This work is built on prime order groups, and the paring operation in prime order group is significant faster than that in composite order groups [23], so the computation overhand is acceptable.

8 Conclusion

This paper designs a new access control system in NDN including the system model, working principle, security definitions and properties in NDN. A new scheme of multi-authority revocable NDN access control based on CP-ABE is implemented by specific security assumption, which solves revocation problem of access control in NDN effectively and protects the privacy of consumers. The attribute-related keys are updated indirectly in each time period, and the keys are renewed periodically by using both the freshness period in the data packet and the in-network caching mechanism of NDN. The revoked consumers cannot obtain the update keys so that user revocation is achieved. Meanwhile, thanks to the decentralization, attribute revocation is easy to be done by the corresponding AAs which control the revoked attributes without any interaction to other AAs. As the access control policy itself will reveal the consumer's sensitive information, we introduce the method of partially hidden access control structure, which will not increase the consumer's key length and guarantee the consumer's privacy. Finally, the security is proved under the static security model.

Acknowledgments

This work is supported by the National Nature Science Foundation of China (No. 61462060), (No. 61461027), the Natural Science Foundation of Gansu Province of China (No. 1610RJYA008). The authors gratefully acknowledge the anonymous reviewers for their valuable comments.

References

[1] N. Attrapadung and H. Imai, "Conjunctive broadcast and attribute-based encryption," in *Third International Conference on Pairing-Based Cryptography (Pairing'09)*, pp. 248–265, 2009.

[2] M. Bayat, M. Aref, "An attribute based key agreement protocol resilient to KCI attack," *International Journal of Electronics and Information Engineering*, vol. 2, no. 1, pp. 10–20, 2015.

[3] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attribute-based encryption," in *IEEE Symposium on Security and Privacy (SP'07)*, pp. 321–334, 2007.

[4] A. Boldyreva, V. Goyal, and V. Kumar, "Identity-based encryption with efficient revocation," in *ACM Conference on Computer and Communications Security*, pp. 417–426, 2008.

[5] M. Chase, "Multi-authority attribute based encryption," in *Conference on Theory of Cryptography*, pp. 515–534, 2007.

[6] M. Chase and S. M. Chow, "Improving privacy and security in multi-authority attribute-based encryption," in *ACM Conference on Computer and Communications Security*, pp. 121–130, 2009.

[7] T. Chen, K. Lei, and K. Xu, "An encryption and probability based access control model for named data networking," in *Performance Computing and Communications Conference*, pp. 1–8, 2015.

[8] H. Cui and R. H. Deng, "Revocable and decentralized attribute-based encryption," *The Computer Journal*, vol. 59, no. 8, pp. 1220–1235, 2016.

[9] C. Ghali, M. A. Schlosberg, G. Tsudik, and C. A. Wood, "Interest-based access control for content centric networks," *Proceedings of the 2nd ACM Conference on Information-Centric Networking (ACM-ICN'15)*, pp. 147–156, 2015.

[10] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data," *Proceedings of the 13th ACM Conference on Computer and Communications Security*, pp. 89–98, 2006.

[11] B. Hamdane and S. G. El Fatmi, "A credential and encryption based access control solution for named data networking," in *IFIP/IEEE International Symposium on Integrated Network Management*, pp. 1234–1237, 2015.

[12] M. A. Hamza, J. F. Sun, X. Y. Nie, Z. Q. Qin, and H. Xiong, "Revocable ABE with bounded ciphertext

Table 2: Characteristics comparison in [8, 13, 23] and this scheme

schemes	Bilinear groups	revocable	Multi-authority	Anonymous of access structure	security
Rouselakis *et al.* [23]	Prime order	×	√	×	Static security
Cui *et al.* [8]	Composite order	Indirect revocation	√	×	Adaptively security
Lai *et al.* [13]	Composite order	×	×	Partially-hidden	Adaptively security
This scheme	Prime order	Indirect revocation	√	Partially-hidden	Static security

Table 3: efficiency comparison in [8, 13, 23] and this scheme

schemes	Public key size	Private key size	Ciphertext size	Paring operations for decryption
Rouselakis *et al.* [23]	$2\|\mathcal{U}_\theta\|$	$2\|\mathcal{S}\|$	$4l+1$	$3\|I\|$
Cui *et al.* [8]	$2\|\mathcal{U}_\theta\|$	$\|\mathcal{S}\|$	$3l+1$	$2\|I\|$
Lai *et al.* [13]	-	$\|\mathcal{S}\|+2$	$2(2l+2)$	$6\|I\|$
This scheme	$2\|\mathcal{U}_\theta\|$	$2\|\mathcal{S}\|$	$2(4l+1)$	$6\|I\|$

in cloud computing," *International Journal of Network Security*, vol. 19, no. 6, pp. 973–983, 2017.

[13] J. Z. Lai, R. H. Deng, and Y. J. Li, "Expressive CP-ABE with partially hidden access structures," in *ACM Symposium on Information, Computer and Communications Security*, pp. 18–19, 2012.

[14] C. C. Lee, P. S. Chung, M. S. Hwang, "A survey on attribute-based encryption schemes of access control in cloud environments," *International Journal of Network Security*, vol. 15, no. 4, pp. 231–240, July 2013.

[15] A. Lewko and B. Waters, "Decentralizing attribute-based encryption," in *Advances in Cryptology (EUROCRYPT'11)*, pp. 568–588, 2011.

[16] Q. Li, J. F. Ma, R. Li, X. Liu, J. B. Xiong, and D. W. Chen, "Secure, efficient and revocable multi-authority access control system in cloud storage," *Computers & Security*, vol. 59, no. C, pp. 45–59, 2016.

[17] Q. Li, X. W. Zhang, Q. J. Zheng, and R. Sandhu, "Live: Lightweight integrity verification and content access control for named data networking," *IEEE Transactions on Information Forensics & Security*, vol. 10, no. 2, pp. 308–320, 2015.

[18] C. W. Liu, W. F. Hsien, C. C. Yang, and M. S. Hwang, "A survey of attribute-based access control with user revocation in cloud data storage," *International Journal of Network Security*, vol. 18, pp. 900-916, 2016.

[19] H. Ma, T. Peng, Z. Liu, "Directly revocable and verifiable key-policy attribute-based encryption for large universe," *International Journal of Network Security*, vol. 19, no. 2, pp. 272–284, 2017.

[20] P. Muralikrishna, S. Srinivasan, N. Chandramowliswaran, "Secure schemes for secret sharing and key distribution using pell's equation," *International Journal of Pure & Applied Mathematics*, vol. 85, no. 5, pp. 933-937, 2013.

[21] T. Nishide, K. Yoneyama, and K. Ohta, "Attribute-based encryption with partially hidden encryptor-specified access structures," in *International Conference on Applied Cryptography and Network Security*, pp. 111–129, 2008.

[22] K. Riad, "Multi-authority trust access control for cloud storage," in *International Conference on Cloud Computing and Intelligence Systems*, pp. 429–433, 2016.

[23] Y. Rouselakis and B. Waters, "Efficient statically-secure large-universe multi-authority attribute-based encryption," in *International Conference on Financial Cryptography and Data Security*, pp. 315–332, 2015.

[24] A. Sahai and B. Waters, "Fuzzy identity-based encryption," in *International Conference on Theory and Applications of Cryptographic Techniques*, pp. 457–473, 2005.

[25] R. S. D. Silva and S. D. Zorzo, "An access control mechanism to ensure privacy in named data networking using attribute-based encryption with immediate revocation of privileges," in *Consumer Communications and Networking Conference*, pp. 128–133, 2015.

[26] X. B. Tan, Z. F. Zhou, C. Zou, Y. K. Niu, and X. Chen, "Copyright protection in named data networking," in *Sixth International Conference on Wireless Communications and Signal Processing*, pp. 1–6, 2014.

[27] Y. Tian, Y. Peng, G. Gao, X. Peng, "Role-based access control for body area networks using attribute-based encryption in cloud storage," *International Journal of Network Security*, vol. 19, no. 5, pp. 720–726, 2017.

[28] R. Tourani, T. Mick, S. Misra, and G. Panwar, "Security, privacy, and access control in information-centric networking: A survey," *Networking and Internet Architecture*, 2016.

[29] C. A. Wood and E. Uzun, "Flexible end-to-end content security in ccn," in *Consumer Communications and Networking Conference*, pp. 858–865, 2014.

[30] G. Xylomenos, C. N. Ververidis, Va. A. Siris, N. Fotiou, C. Tsilopoulos, X. Vasilakos, Konstantinos V. Katsaros, and G. C. Polyzos, "A survey of information-centric networking research," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 2, pp. 1024–1049, 2014.

[31] L. X. Zhang, A. Afanasyev, J. Burke, V. Jacobson, K. Claffy, P. Crowley, C. Papadopoulos, L. Wang, and B. C. Zhang, "Named data networking," *ACM Sigcomm Computer Communication Review*, vol. 44, no. 3, pp. 66–73, 2014.

[32] Y. Zhao, P. Fan, H. Cai, Z. Qin and H. Xiong, "Attribute-based encryption with non-monotonic access structures supporting fine-grained attribute revocation in M-healthcare," *International Journal of Network Security*, vol. 19, no. 6, pp. 1044–1052, 2017.

Biography

Tao Feng received his D.E from Xidian Univerisity in 2008. He is now a professor and doctoral supervisor in Lanzhou Univerisity of Technology. His research interests include network and information security.

Jiaqi Guo received the B.E in information security from Xi'an University of Posts and Teleconmmunications in 2012. She is currently a postgraduate student in Lanzhou University of Technology. His current research interests include cryptography and network security.

Defense Techniques of SYN Flood Attack Characterization and Comparisons

Shaila Ghanti, G. M. Naik
(Corresponding author: Shaila Ghanti)
Department of Electronics, Goa University
Taleigao Plateau, Goa 403206, India
(Email: shailaghanti@yahoo.com)
(Received Mar. 26, 2017; revised and accepted June 26, 2017)

Abstract

Automation systems are being used widely for performing different tasks at homes, offices, and industries. Remote clients access these web based automation system services on the Internet. Web services are also prone to attacks due to SYN flood that deny genuine clients a right to use the services. Therefore such services are required to be secured. The primary purpose of this manuscript is to characterize the defense mechanisms and compare the technical details involved in defense mechanisms of attacks due to SYN flood. This will help the researchers to propose improved and more efficient defense mechanisms. The manuscript furthermore includes the experimental study of the Victim Side SYN Flood (VSSF) attack protection system implemented using a general purpose processor during the attack. Subsequently the study compares the performance of VSSF attack protection system implemented using general purpose processor, with the VSSF attack protection system implemented using NIOS core processor. It is found that the NIOS core processor based protection system performance is better than the general purpose processor based systems.

Keywords: Attacks; Defense Mechanisms; FPGA; NIOS Processor; SYN Flood Attack

1 Introduction

Today a vast number of online services are used to transfer crucial data. These online services are widely used by the users to transfer data. Meanwhile the basic principles like integrity, confidentiality, and availability of resources have to be applied to secure these services.

Distributed Denial of Service (DDoS) attack is generated on the network to deny access to the services by utilizing the resources of the network or system [2, 4, 14, 18]. DDoS attacks are classified into many types of attacks, such as UDP flood, SYN flood, Ping flood attacks, etc. [3, 7, 25] . In this study we have concentrated on SYN flood attacks. SYN flood attack is a type of DDoS attack that will target the specific resources of victims [29]. These attacks generated on the server will disrupt the services and genuine users are not able to access these services. Therefore, there is a need to protect such services from attacks. Although there are many types of defense systems against the SYN flood attacks, contributed by the researchers are available, designed using software [5] and hardware [10], still the issue remains challenging.

This manuscript deals with the characterization and classification of defense mechanisms used only for SYN flood attack. This will help the researchers to propose improved and more efficient defense mechanisms. Furthermore it proposes the Victim Side SYN Flood (VSSF) attack protection system implemented using a general purpose processor.

Subsequently the performance of VSSF attack protection system implemented using general purpose processor is compared, with the VSSF attack protection system implemented using NIOS core processor [10]. This comparison will help to choose and design the appropriate security system.

Contributions of this manuscript are:

1) Identify the characteristics of various defense mechanisms against attacks due to SYN flood.

2) Compare the various defense systems, which will help researchers to invent a comprehensive and efficient defense system so that the genuine user's access to server is not denied.

3) Experimental study of proposed VSSF attack protection system implemented on a PC using a general purpose processor.

4) Compare the performance of the general purpose processor based system, with the NIOS core processor based VSSF protection system.

2 SYN Flood Attack

To communicate over the network between any server and client using TCP, the client has to first set up a connection with the server. Then they can receive and transmit the data. After the data transfer is done, the connection has to be disconnected. The process of setting up of the connection between client and server involves exchange of three way handshake signals [23]. The client first sends the SYN packet to the server, the server in turn sends the SYN-ACK back to the client by setting the TCP half open connection on the server. Then the ACK is sent from the client side to the server and the connection is set up between the client and server in normal situation.

In order to disrupt the Internet services available on the server the malicious attacker can generate the SYN flood attack. During this attack the attacker attacks the server by sending many spoofed SYN packets. Due to these packets the TCP half open connections are set up on the server and then the server sends back SYN-ACK to the clients and waits for the corresponding ACK [15, 22]. As IP addresses used are spoofed there may not be a client with that IP address and subsequently the server will not receive corresponding ACK packet. Each TCP half open connections reserve certain resources of server. Large number of SYN flood attack leads to the consumption of server resources and genuine users will not have access to the server.

3 SYN Flood Attack Defense Mechanism Classifications and Their Characterizations

Literature review shows that many types of defense mechanisms are contributed by the researchers. Here we characterize and compare the different mechanisms.

3.1 Characterization

Defense mechanisms of SYN flood systems are classified based on the location of deployment. The defense mechanisms can be deployed at the source side, victim side and intermediate on the network side.

Source side defense systems: These are the defense systems implemented at the attacker side. These defense systems are more effective as the attacks are blocked at the source side itself. However these defense systems need to be deployed at all the source/client side.

Victim side defense systems: Victim side defense systems can detect and defend the attack easily as all the attack requests are available at the victim side. It is sufficient only to deploy the defense system near the victim.

Intermediate defense system on the network: Generally the defense systems are deployed in between the source and the victim. Here detecting the attack is difficult as all the packets of attack are not available at one place.

Defense mechanism systems are classified as Distributed or Autonomous. The autonomous systems independently take decision to protect the servers. Whereas distributed systems needs coordination amongst different systems to protect the servers.

3.2 Comparisons

The comparison of the following different defense mechanisms of attack due to SYN flood is shown in Table 1.

SYN Cookies: In this technique no memory is reserved to store the initial request information on server. Instead a code is generated using the received initial request information and cryptographic techniques. This code is used as "sequence number" in the SYN-ACK packet and sent back. When the respective ACK is received it extracts the initial request information and is used to set up the connection [5].

SYNkill: Detects the attack due to SYN flood and then responds to lessen the effect of attack. SYNkill tool generates RST and ACK packets depending on the type of client request so that the resources of the server are not wasted [24].

Ingress Filtering: It is a source side defense system. It will only forward the packets that have the address of source(Prefix) same as the source network address(prefix). The exit or gateway routers are configured in such a way that they block all packets that are not the component of the input network address [9]. It is more effective provided, it is implemented for all clients.

SYN Cache: In this method the resource allocation strategy is changed. Minimum initial information of the request is stored and then during the connection all the required information is stored [16].

SYNMON: Network processor is used as a processing unit to detect the SYN attack. An embedded system is designed to detect the attack using the CUSUM method [17].

D-WARD: This technique is implemented near the source, but also supports near the victim and within the core. However the source end deployment supports good response as compared to victim end and core side. The D-WARD scheme consists of 3 units called observation unit, traffic policing unit and rate-limiting unit. It maintains the information of all the traffic and then the statistics of aggregate flow is periodically compared with the models of traffic to detect the attack. Attack Mitigation is performed by the method called rate limiting [19].

Throttling Source Side SYN Flooding Attack: It is a defense mechanism that needs to be deployed at the attacker/source side. Bloom filter is used to store the information of traffic as it provides storage

efficient data structure. CUSUM method detects the malfunctions in the traffic. Thus identifies the attack and then the mitigation of attack is performed using the ingress filter and rate limiting scheme [6].

A Router-based Novel Scheme: This method detects and mitigates the attack due to SYN flood. It is a router based mechanism that uses Bloom filter (counting) to keep track of number of SYN and FIN/RST packets. During the attack,if the count of SYN is more than number of FINs [26], then the attack is detected. During mitigation every client's first request is dropped. The client retransmits the request and only such requests are forwarded to server. Thus the effect of attack is mitigated.

Active Probing Method: Attack detection is done using the probing technique. Based on the values of TTL the Rate-limiting filter is used to mitigate the attack [28].

Detection of IP Header Threats Using Anomaly Detection: Anomaly detection methods use the information from headers of TCP and IP to detect the attack. Traffic capturing tools are used to capture the TCP and IP information on the network. Anomaly detection supports three types of detection methods. The first type is the protocol detection wherein any violations in the protocol pattern is detected. The second type is based on the rate that detects the attack by comparing it with the normal traffic rate and the third type is based on behavioural changes that compares with the normal behaviour of clients with the servers. In this rate based detection is used. Once the attack is detected attentive messages are sent to the administrator [13].

STONE: This method presents a mechanism with expert system functions that protects the server from DDoS attack. STONE detects the attack and mitigates the attack. The network traffic is captured and it aggregates all the addresses into common prefix of IP addresses. The attack is identified when these aggregated data deviates from the regular traffic flow. Following the attack discovery the STONE allows traffic from known sources to access the services where as the other traffic is blocked [12].

Secured Ethernet Interface System: This method detects and mitigates the attack. It classifies the incoming requests as good by authenticating. Every incoming request with the good IP is forwarded and other requests are blocked. During spoofed SYN flood attack it allows only good requests to be passed to the server and blocks all other attacks. However it is likely that the attack requests with the spoofed good IPs are forwarded to the server. Once the number of half open connections becomes greater than the threshold value then the good registry is cleared. Since the good registry is cleared every incoming request is authenticated and then only it forwards. Thus it blocks the attacks [10].

3.3 Parameters To Be Measured

SYN flood attack defense systems performance can be analysed based on the below mentioned parameters that need to be measured by the researchers.

1) Response time: It gives the total time period needed by the server(web server) to react to client request. In linux the tool called time curl, ab tool (linux/windows) can be used [21].

2) Connections and half-open connections: Total half-open and complete connections setup on the server is an important parameter. The connections information indicates the number of genuine client requests served by the server. To measure these parameters Netstat tool is used.

3) Processor utilization: It is used to measure the quantity of processing achieved by the server processor. Top tool can be used to measure the utilization of processor.

4) Network bandwidth: The bandwidth of the network that is protected is to be measured. This indicates the traffic details.

The response time, number of TCP connections, half open connections, processor utilization and network bandwidth parameters can be measured in three different scenarios while genuine requests are sent to the server.

1) Without generating attack and no protection system;

2) With attack but no protection system;

3) With attack as well as protection system.

4 VSSF Protection System

In Secured Ethernet Interface System [10] the protection against attack is performed using the NIOS II soft core processor. It is interesting to compare the performance of the SYN flood attack protection system using PC based pure software and FPGA based NIOS processor. This information is very useful to decide upon what type of security system must be used for protecting the services.

The VSSF protection system is proposed and implemented using the general purpose processor on a PC. This system is implemented using the same algorithm as proposed in Secured Ethernet Interface System and the performance is analysed experimentally.

4.1 VSSF Protection Method

Figure 1 indicates the VSSF protection method used for protecting the SYN flood attack protection system as used

Table 1: Comparison of SYN flood attack defense methods

Year	Method name	Deployment	Method used	Spoofing	Testing (Simulations/ Real time)	Distributed/ Autonomous	Testing data	Software/ hardware	Results/ parameters measured
1996	SYN cookies	Victim	No need of memory to store initial request information	Yes	——	Autonomous	—	Software	Implemented in Linux
1997	SYNkill	Victim	Classification of client requests	Yes	Real time	Autonomous	—-	Software	Delay in setting up of connections and number of connections
2000	Ingress Filtering	Source	Filtering	Yes	—-	Autonomous	—	Software	—
2002	SYN Cache	Victim	Initial minimum information is stored	—	Real time	Autonomous	—-	Software	Percentage of connections set up and time taken during the attack
2005	SYNMON	Victim	CUSUM	yes	Realtime	Autonomous (Detection)	Packt, hping	Hardware/ Network Processor	Attack detection
2005	D-WARD	source	Aggregate flow statistics and compare with models of traffic	Yes	Emulab	Autonomous/ distributed	Cleo tool	Software	Number of connection, connection delay, failed connections
2006	Throttling SYN flooding attack	Source	Bloom filter, CUSUM method, ingress filter	yes	Simulations (ns2 simulator)	Autonomous	DARPA dataset	Software	Number of connections set up and detection rate
2007	Router based Novel scheme	Victim	Counting Bloom filter to keep track of SYN and FIN/RST. Persistence of client property	Yes	Simulations (Trace driven)	Autonomous	—	Software	Keeps track of number of SYNs and FIN/RSTs
2008	Active Probing technique	Victim	Active probing and Ratelimiting method	Yes	Simulations (NS2-simulator)	Autonomous	Simulations	Software	Bandwidth consumption of server by legitimate traffic
2011	Anomaly Detection	Vctim	Rate based Anomaly Detection	—-	Real time	Autonomous (Detect)	tcpdump	software	Packets are tested
2015	STONE	Distributed	Streaming processing paradigm	—	Real time	Distributed	—	Software	Attack detection time and Mitigation precision
2016	Secured Ethernet Interface System	Victim	Keeping track of the genuine requests	yes	Real time	Autonomous	Ostinato and ab tools are used	FPGA, NIOS (Hardware)	Measured half open connections, connections set up

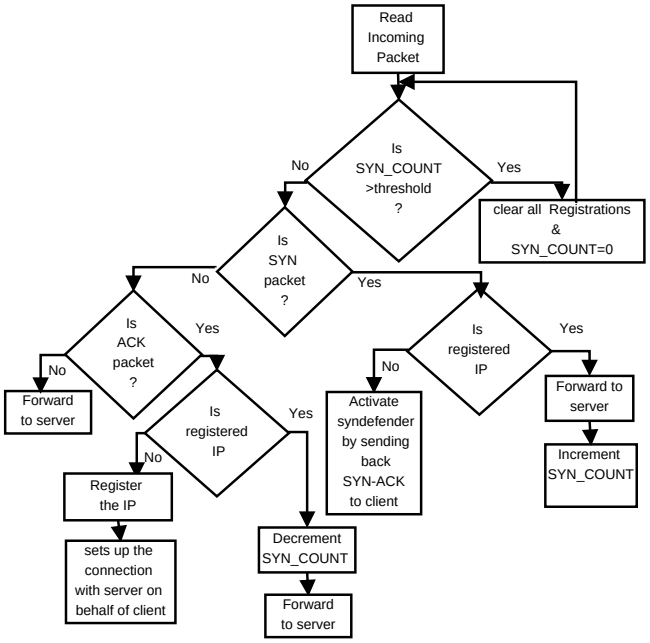


Figure 1: VSSF protection method [10]

by S.Ghanti and G.M.Naik [10], and is based on [11, 24, 27].

This method detects the spoofed attack and also it blocks such attacks. In this method a registry is used that maintains the information about the IP addresses of clients that have already accessed and set the connection with the server. Every incoming client request needs to be identified by the VSSF protection system as a genuine client request or from an attacker and is explained below.

The IP address of the incoming client request is first compared with the contents of the registry. If the entry is not traced in the registry then these are termed as new clients or non registered clients. These new clients are not forwarded to the server. Instead the SYN-ACK packet is sent to the client using the syndefender. If this request had come from the genuine client then corresponding reply is received by the VSSF protection system and will set up the connection between the client and the server. Also entry in the registry is added with such request information. If the request had come from 'non registered attack clients' then the VSSF protection system will not receive the ACK thus they are treated as attack requests and are not forwarded instead they are blocked.

An incoming client request if found in the registry, then it is treated as registered client request. The registered client requests are forwarded to the server and the server communication continues in a normal way.

The VSSF protection system uses the below explained method to detect the spoofed client requests attack. The spoofed client requests generated can be either from registered or non registered IP addresses. The non registered spoofed requests are treated as new client requests and taken care by not forwarding to the server and activating the syndefender. Thus non registered spoofed requests are taken care by the VSSF protection system.

In case of registered spoofed request attacks, the VSSF protection system uses an innovative method to detect and block the spoofed attacks. On receiving the spoofed registered request the VSSF protection method forwards such packets to the server as they are registered. For such packets SYN_Count is incremented and the SYN_ACK is sent back. Since it is a spoofed registered request, ACK is not received thus the SYN_Count will not be decremented. Thus for every registered spoofed request the SYN_Count goes on increasing. The spoofed attack is detected once the count reaches the threshold value. Once it detects the attack further attack is blocked by clearing the registry. Thus the VSSF protection system detects the attack and it also blocks the attack.

4.2 Software-based VSSF Protection System

To implement the software based VSSF protection system a PC with two Network Interface Cards (NIC) is used. This system needs to be connected between the server that needs to be secured and the clients. To start with, this system is configured using iptables as router so that it forwards the packets. The VSSF protection system should be able to capture every incoming packet and analyse it using the same algorithm as used in FPGA based Secured Ethernet Interface System shown in Figure 1 to detect and block the attack [10]. The VSSF protection system software is implemented using **libipq** so that it analyses every incoming packet and accordingly the packet is forwarded or blocked [8]. **Libipq** is an iptables packet queing development library at the user space. The libipq provides different APIs to create handle, to set mode, to read packets from queue, to issue verdict on packet like drop, forward, to destroy the handle etc. It allows packets to pass to the user space where packet details can be analysed. Then the packets can be passed to the kernel indicating whether packet can be dropped or forwarded to the server. The packet contents can also be modified if required and then the packets can be passed to the kernel from user space.

Libipq APIs are used to monitor the incoming requests from the clients/attackers and identify the SYN flood attacks and accordingly block the SYN flood attacks is shown in Figure 1 so that server resources are not consumed by the attacks, and also genuine users will have access to the server.

The set up used for the software based experiment is shown in Figure 2. One computer is configured as a web server, while another is used as a client to generate the attacks and the genuine requests. The third machine is the VSSF protection system implemented using software. We have used apache bench ab tool to generate genuine client requests to the server [1] and Ostinato tool to generate SYN flood attack to the server [20]. Experiments were conducted to study the response of VSSF protection system with and without attack.

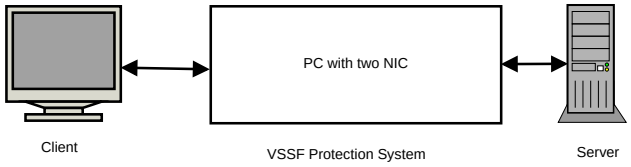


Figure 2: The software based VSSF protection system experimental set up

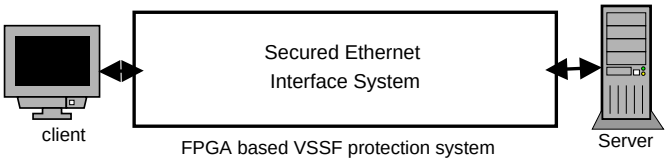


Figure 3: FPGA based protection system

4.3 FPGA-based Protection System

The FPGA based Secured Ethernet Interface System (referred here as FPGA-based VSSF protection system) was implemented on a hardware DE-4 FPGA board as a System on Chip and was reported in [10] by Ghanti and Naik. It uses Stratix IV device supported by the industry standard peripherals. Different IP cores used are Triple-speed Ethernet core IP, receive and transmit SGDMA, NIOS processor, on chip memory, etc. [10]. SYN flood attack protection system flowchart is as shown in Figure 1 and was implemented using Quartus II.

The experimental setup used in FPGA based Secured Ethernet Interface System [10] is shown in Figure 3. One computer was configured as a web server, while another was used as a client to generate the attacks and the genuine requests.

The FPGA based protection system was connected just before the server. Experiments were performed to study the response of VSSF protection system with and without attack. The results were discussed in [10] by S.Ghanti and G.K.Naik.

5 Results and Discussions

5.1 Results of Software-based VSSF Protection System

To find out the performance of software based VSSF protection system, studies are conducted initially by generating genuine requests from the client using ab tool (without generating attack) to the server and the response is noted. In this experiment ab tool is used to generate 500 genuine requests to the server.

Later the client generates genuine requests with the attacks to the server . The attacks are generated using ostinato tool to the server. Then the response of the server is recorded and is shown in Figure 4 and Table 2.

From Figure 4 it is clear that 90 % of the requests from client are served within less time when no attack is

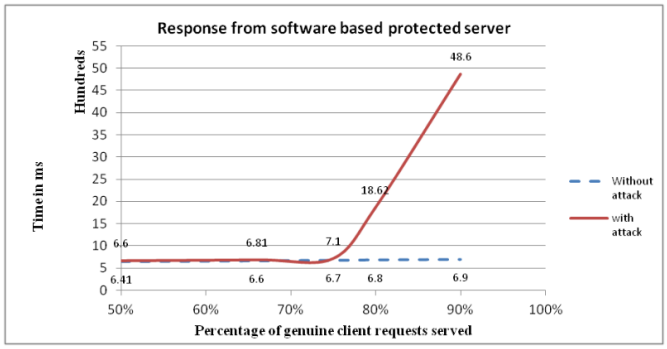


Figure 4: Response from software based VSSF protected server

Table 2: Response of software based VSSF protection system with and without attack (when client sends 500 genuine requests).

	Using PC based protection system without attack	Using PC based protection system with attack
Transfer Rate (Kilobytes/sec)	0.49	0.19
Time taken for tests (in seconds)	315	825

generated. During the attack, genuine clients could access the server but only the delay is more. It may be seen in Figure 4 that there is a sharp change at around 75% of the client requests i.e. response time of server quite significant when there is an attack. Thus, the software based VSSF protection system protects the server from the attack and also genuine clients can access the server during attack.

5.2 Comparison of Software(PC)-based With The FPGA-based Protection System

The experimental results of the PC based protection system are shown in Figure 4. The results of the experiments conducted by generating attacks and genuine clients' requests to the server that is protected by FPGA based SYN flood attack protection systems as reported by S.Ghanti and G.K.Naik in [10] is used here for comparison. The comparison of results of software based VSSF protection system, with the FPGA based protection system are tabled in Table 3 and are shown in Figure 5.

In Figure 5 there is a sharp change at 75% of client requests when the server is protected using software based system, but the response of hardware i.e. FPGA based VSSF protection system shows much better response as

Table 3: Comparison of FPGA based protection system and software based VSSF protection system with attack (500 Genuine Requests with attack are sent from client to the server).

	Using PC (software) based protection System	Using FPGA based protection system [10]
Transfer Rate (Kilobytes/sec)	0.19	2.25
Time taken for tests(in seconds)	825	64.179

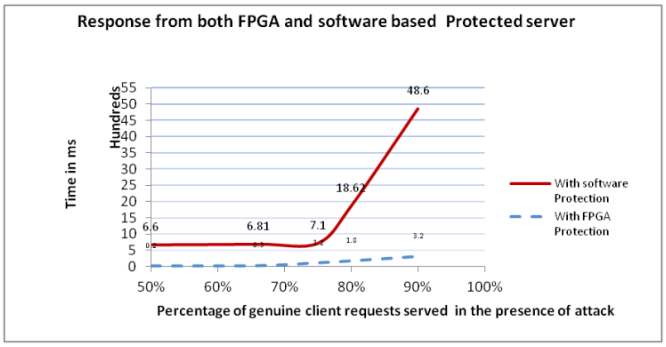


Figure 5: Comparison of FPGA based and software based protection system

compared to software based in spite of an attack. Figure 6 indicates that the FPGA (NIOS) based VSSF protection system supports higher transfer rate as compared to software based VSSF protection system.

6 Conclusions

SYN flood attacks generated on the servers cause the services to be disrupted. Defense systems reported in the literature are characterized and compared extensively. Different parameters of defense system that should be measured experimentally are suggested. This will help researchers to develop better efficient systems.

This article also demonstrates the implementation and working of the VSSF protection system on PC experimentally. It then compares the performance of PC based with the FPGA based System on Chip protection system. It is found that VSSF protection system implemented using software efficiently blocks the attack and allows genuine requests to access the server. The FPGA based SYN flood attack protection system supports the faster data transfer than the software based system. Thus the protection system designed using FPGA is more efficient than the

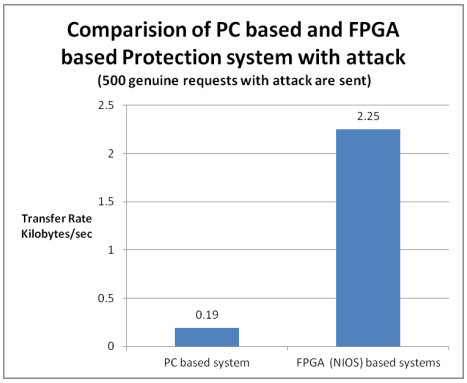


Figure 6: Comparision of PC (software) based and FPGA based VSSF protection system with attack

software based protection system.

Though the FPGA based SYN flood attack protection system shows better results as compared to PC based systems, it may be noted that FPGA solutions are not easy as one requires access to hardware, whereas PC based system provides a solution which can be implemented and administered locally very easily. Further the response of software based VSSF protection system can be easily improved by incorporating parallel processing.

Acknowledgments

The authors gratefully acknowledge the anonymous reviewers for their valuable comments.

References

[1] Apache, *ab - Apache http Server Benchmarking Tool*, Jan. 12, 2018. (`https://httpd.apache.org/docs/2.4/programs/ab.html`)

[2] R. C. Baishya, N. Hoque, and D. K. Bhattacharyya, "DDoS attack detection using unique source IP deviation," *International Journal of Network Security*, vol. 19, no. 6, pp. 929–939, 2017.

[3] S. Behal and K. Kumar, "Characterization and comparison of DDoS attack tools and traffic generators: A review," *International Journal of Network Security*, vol. 19, no. 3, pp. 383–393, 2017.

[4] S. Behal, K. Kumar, M. Sachdeva, "Discriminating flash events from DDoS attacks: A comprehensive review," *International Journal of Network Security*, vol. 19, no. 5, pp. 734-741, 2017.

[5] D. J. Bernstein, *SYN Cookies*, Jan. 12, 2018. (`http://cr.yp.to/syncookies.html`)

[6] W. Chen and D.-Y. Yeung, "Throttling spoofed SYN flooding traffic at the source," *Telecommunication Systems*, vol. 33, no. 1, pp. 47–65, 2006.

[7] S. Deore and A. Patil, "Survey denial of service classification and attack with protect mechanism for TCP SYN flooding attacks," *International Research*

Journal of Engineering and Technology, vol. 3, no. 5, pp. 1736–1739, 2016.

[8] die.net, *libipq(3) - Linux Man Page*, Jan. 12, 2018. (URLhttps://linux.die.net/man/3/libipq)

[9] P. Ferguson, D. Senie, *Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing*, RFC 2827, 2000.

[10] S. R. Ghanti and G. Naik, "Efficient data transfer rate and speed of secured ethernet interface system," *International Scholarly Research Notices*, vol. 2016, 2016.

[11] S. R. Ghanti and G. Naik, "Protection of server from SYN flood attack," *International Journal of Electronics and Communication Engineering & Technology*, vol. 5, no. 11, pp. 37–46, 2014.

[12] V. Gulisano, M. Callau-Zori, Z. Fu, R. Jiménez-Peris, M. Papatriantafilou, and M. Patiño-Martínez, "Stone: A streaming DDoS defense framework," *Expert Systems with Applications*, vol. 42, no. 24, pp. 9620–9633, 2015.

[13] S. Haris, G. M. W. Al-Saadoon, A. P. D. R. Ahmad, and M. Ghani, "Anomaly detection of IP header threats," *International Journal of Computer Science and Security*, vol. 4, no. 6, p. 497, 2011.

[14] N. Hoque, D. K. Bhattacharyya, and J. K. Kalita, "FFSC: A novel measure for low-rate and high-rate DDoS attack detection using multivariate data analysis," *Security and Communication Networks*, vol. 9, no. 13, pp. 2032–2041, 2016.

[15] D. Kshirsagar, S. Sawant, A. Rathod, and S. Wathore, "CPU load analysis & minimization for TCP SYN flood detection," *Procedia Computer Science*, vol. 85, pp. 626–633, 2016.

[16] J. Lemon, "Resisting SYN flood dos attacks with a SYN cache," in *Proceedings of the BSD Conference (BSDC'02)*, pp. 89–97, 2002.

[17] B. Lim and M. S. Uddin, "Statistical-based syn-flooding detection using programmable network processor," in *Third International Conference on Information Technology and Applications*, vol. 2, pp. 465–470, 2005.

[18] J. Mirkovic and P. Reiher, "A taxonomy of DDoS attack and DDoS defense mechanisms," *ACM SIGCOMM Computer Communication Review*, vol. 34, no. 2, pp. 39–53, 2004.

[19] J. Mirkovic and P. Reiher, "D-ward: A source-end defense against flooding denial-of-service attacks," *IEEE Transactions on Dependable and Secure Computing*, vol. 2, no. 3, pp. 216–232, 2005.

[20] Ostinato, *Ostinato Network Traffic Generator Network Traffic Generator and Analyzer*, Jan. 12, 2018. (http://ostinato.org/)

[21] S. Rao and S. Rao, "Denial of service attacks and mitigation techniques: Real time implementation with detailed analysis," *SANS Institute Reading Room*, 2011.

[22] H. Safa, M. Chouman, H. Artail, and M. Karam, "A collaborative defense mechanism against SYN flooding attacks in IP networks," *Journal of Network and Computer Applications*, vol. 31, no. 4, pp. 509–534, 2008.

[23] S. Sathwara, C. Parekh, "Distributed denial of service attacks – TCP SYN flooding attack mitigation," *International Journal of Advanced Research in Computer Science*, vol. 8, no. 5, pp. 2392–2396, 2017.

[24] C. L. Schuba, I. V. Krsul, M. G. Kuhn, E. H. Spafford, A. Sundaram, and D. Zamboni, "Analysis of a denial of service attack on TCP," in *Proceedings of IEEE Symposium on Security and Privacy*, pp. 208–223, 1997.

[25] A. Srivastava, B. Gupta, A. Tyagi, A. Sharma, and A. Mishra, "A recent survey on DDoS attacks and defense mechanisms," in *Advances in Parallel Distributed Computing*, pp. 570–580, 2011.

[26] C. Sun, J. Fan, L. Shi, and B. Liu, "A novel router-based scheme to mitigate SYN flooding DDoS attacks," *IEEE INFOCOM (Student Poster)*, 2007.

[27] C. Sys, *Check Point Software Techs., Inc. v. SRI Int'l, Inc.*, Jan. 12, 2018. (https://www.casemine.com/judgement/us/5914e6c3add7b04934911071)

[28] B. Xiao, W. Chen, and Y. He, "An autonomous defense against SYN flooding attacks: Detect and throttle attacks at the victim side independently," *Journal of Parallel and Distributed Computing*, vol. 68, no. 4, pp. 456–470, 2008.

[29] S. T. Zargar, J. Joshi, and D. Tipper, "A survey of defense mechanisms against distributed denial of service (DDoS) flooding attacks," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 4, pp. 2046–2069, 2013.

Biography

Shaila Ghanti is a Ph.D. student in the Department of Electronics Goa University, Goa. She is teaching in the Department of Computer Science in Chowgule College, Margao Goa, India. She is an author for few research paper and completed two minor research project. Her research area is networking, network security and embedded Systems.

Prof. Gourish Naik obtained his Ph.D from Indian Institute of Science, Bangalore (1987) and served the institute as research associate in the areas of Optoelectronics and Communication till 1993. He was also senior research Fellow of BARC for the period 1985-1987. For the last 15 years, he is associated with Goa University Electronics Program. He is the founding head of University Instrumentation Center of Goa University. He is also coordinator of DEITI (an educational broadcast studio supported by Indian Space Research). His other commitments are regulating digitization Center at Goa University to support the various Digital repository projects like DIGITAP (Digital Repository for Fighter Aircrafts of Indian Navy) Million Book project of Ministry of Information Technology, New Delhi and Antarctica Study Center (NCAOR), Govt.of India. He has to his credit around 50 odd research

papers published in National and International Journals and has presented research works at various National and International Forums. He has delivered several keynote addresses and invited talks at various institutes and also authored two books on Embedded Systems published by Springer (Holland). He is a member of Goa State Rural Development Authority, member of advisory board, Goa Police Cyber Crimes and also advisor for Directorate of Technical Education. He was the chairman of Goa University Technical Advisory Committee. Presently he is head of Electronics department at Goa University.

Efficient Anomaly Intrusion Detection Using Hybrid Probabilistic Techniques in Wireless Ad Hoc Network

K. Murugan[1,2] and P. Suresh[3]

(Corresponding author: K. Murugan)

Research Scholar, Bharathiar University[1]

Coimbatore, Tamil Nadu 641046, India (Email: mkcsresearch@gmail.com)

Department of Computer Science, Government College for Women, Kolar-563101, Karnataka, India[2]

Department of Computer Science, Salem Sowdeswari College, Salem 636010, Tamil Nadu, India[3]

(Received Mar. 26, 2017; revised and accepted June 26, 2017)

Abstract

A wireless ad-hoc network includes huge number of mobile nodes that structure temporary networks. Due to the dynamic nature of wireless ad-hoc network, security and efficient intrusion detection system (IDS) is a challenging task to detect the intruder nodes. The classification algorithm is used to detect the intrusions in an efficient manner. However, the network is characterized by high mobility they also introduce many vulnerabilities that increase their accurate detection risks. The optimization technique is performed to attain effective model for intrusion detection. But, the IDS continuously use additional resources to monitoring intruder activity in the network. In order to overcome the above issues in wireless ad-hoc network, Simulated Annealing based Naive Bayes Classifier (SA-NBC) technique is proposed for Anomaly Intrusion Detection. An anomaly-based intrusion detection system is used to detect the network intrusions and monitoring network activities in an exact manner. At first, the optimal features are chosen for classifying and detecting the intrusion by means of Simulated Annealing (SA) method when performing packet transmission. Based on these selected features, the accuracy and efficiency of traffic pattern analysis is improved using intrusion detection. Next, the Naive Bayes classifier is employed to classify the attack depends on features to identify the malicious behavior accurately from normal node in a testing environment by using the Bayes theorem. This in turns, the network traffic is minimized and increases the accuracy of anomaly intrusion detection. The SA-NBC technique conducts the simulations work on parameters such as anomaly intrusion detection accuracy, execution time, and throughput. The simulation results demonstrate that the SA-NBC technique is able to improve the accuracy of intrusion detection and also improves the throughput when compared to state-of-the-art works.

Keywords: Anomaly Intrusion Detection; Bayes Theorem; Intrusion Detection System (IDS); Naive Bayes Classifier; Wireless Ad-Hoc Network

1 Introduction

A wireless ad hoc network (WANET) is a decentralized network without having any fixed infrastructure. Wireless ad-hoc networks are essentially used in the tactical battlefield, emergency explore and civilian ad-hoc locations. The nodes in wireless ad-hoc network communicate with each other nodes through the intermediate node. Due to the diverse characteristic of mobile ad hoc networks, the various intrusions affect the network performance thus increases the traffic. In addition, limited transmission range of wireless network introduces the communication traffic over a number of nodes.

A novel IDS called as Adaptive Three Acknowledgements (A3ACKs) was introduced in [17] for MANETs. In A3ACKs, watchdog technique was applied to solve the issues in data transmission. However, the packet delivery ratio was not improved. In [15], a novel intrusion-detection system called as Enhanced Adaptive ACKnowledgment (EAACK) was developed to increase the malicious-behavior-detection rates in MANETs. However, the network overhead was high and the normal or anomalous behavior activities are difficult to identify.

A new method called as hash message authentication code (HMAC) was introduced in [4] to overcome the primary user attack in cognitive radio networks. However, the interferences are occurred in the HMAC through the transmission process. Security assurance process properties unification was developed in [12] to solve the security demands when handling logical vulnerability in system.

However, some network issues are not exposed or identified.

In [16], Intrusion detection systems were introduced to solve the availability attacks in ad-hoc networks. However, the performance of the network was not improved. A hybrid detection system called Hybrid Intrusion Detection System (H-IDS) [13] was developed to detect the DDoS attacks. The H-IDS uses both anomaly-based and signature-based detection methods. However, the detection accuracy of intrusion detection is not efficient.

In [1], a new intrusion detection system was introduced based on neuro-fuzzy classifier for avoiding the packet drops in mobile ad hoc networks. Anomalous node in network is isolated from the normal activities by using SVM classifier. An IDS based on anomaly based intrusion detection was developed in [8] by protecting the network node behavior to overcome the attacks. However, the packet delivery ratio was not sufficient.

Intrusion detection in MANETs using statistical classification algorithms [10] was developed to improve the classifier performance. A normalized gain based IDS was introduced in [19] for MAC Intrusions (NMI) detection to choose an optimal feature subset in training the classifier. However, the time consumed for classification was high.

The contribution of the paper is organized as follows: Simulated Annealing based Naive Bayes classifier (SA-NBC) technique is developed in wireless ad-hoc network for Anomaly Intrusion detection. Initially, with the aid of Simulated Annealing (SA) in SA-NBC technique, the optimal features of the nodes are selected to classify the intrusion in packet transmission. Next, by considering optimal feature selection, the traffic pattern accuracy is improved in the network. Finally, NB classifier is used to classify the node whether it is normal or anomalous by using the conditional probability function and therefore improves the intrusion detection accuracy and reduces the execution time for intrusion detection.

The rest of the paper is organized as follows. Detailed description of the method is provided in Section 2. In Section 3 the simulation environment is presented and the results are explained in Section 4. Section 5 presents a brief introduction of related works. Finally, the concluding remarks are presented in Section 6.

2 Methodology

Due to the higher mobility of nodes in wireless ad-hoc network, the different intrusions are occurred at the network transmission. A network intrusion is also called as the unauthorized activity on a computer network. The aim of intrusion detection is identify the various types of misbehavior activity. This misbehavior activity utilizes the network resources and accesses the data through the transmission for reducing network performance. Generally efficient modeling and organizing a network intrusion detection system is used to detect the intruders from the network. Therefore, an efficient IDS uses Simulated Annealing based Naive Bayes Classifier (SA-NBC) technique for anomaly intrusion detection in wireless ad-hoc network.

2.1 System Model

In this section, a system model for designing with Naive Bayes classifier is presented. Let us consider a WANET with number of node, $N_i = N_1, N_2, \cdots, N_n$, distributed in a given rectangular area.

2.2 Problem Definition

The major challenging problem in wireless ad-hoc network includes the lack of reliable data transmission due to its mobility and hence it is more prone to intrusion threats. With increases of anomalous in ad-hoc network leads to degrade the network performance. The anomaly intrusion detection is a basic issue for intrusion detection in ad-hoc network. The problem of classify the normal and abnormal nodes during the data transmission is considered in this work with the aim of obtaining improved intrusion detection accuracy using Simulated Annealing based Naive Bayes classifier technique.

Generally, intrusion detection techniques are difficult to distinguish the activities whether it is normal or anomalous. However the network resources are utilized by malicious activity and unable to compromise a classification. Hence, efficient intrusion detection system (IDSs) is needs to increase the throughput of network by means of classifying behavior of the nodes.

2.3 Simulated Annealing Based NB Classifier

A Simulated Annealing based NB classifier (SA-NBC) technique is developed with the aim of increasing the anomaly intrusion detection accuracy in wireless ad-hoc networks. Due to the different features of node the computational complexity is occurred in the network. In order to overcome the above issues, the SA-NBC technique uses the simulated annealing (SA) method that extracts the optimal feature. Each feature contains the different values in data packets that are produced by different attacks. The group of feature values is separated into various classes. Hence, an optimal feature is selected efficiently and reduces the computation time of IDS thus improves the detection accuracy using classifier.

The NB classifier is used to classify the malicious behavior accurately using SA-NBC technique. It reduces the failure of unidentified process and mainly employs the classification task that directs to lack of cascade classification in the intrusion classification process. Therefore, the NB classifier performs an intrusion classification process by means of separating the optimal features through monitoring the malicious activities in precise manner. The architecture of SA-NBC technique for anomaly intrusion detection and classification is shown in Figure 1.

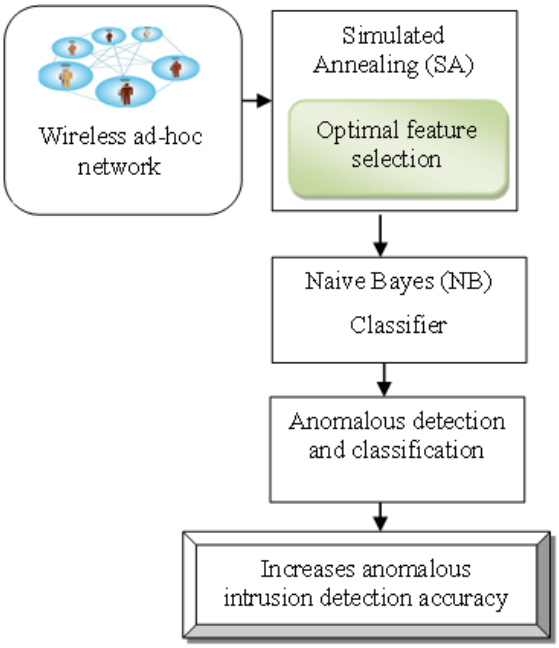


Figure 1: Architecture diagram of simulated annealing based NB classifier

As shown in Figure 1 the process of Simulated Annealing based NB classifier is used to detect the intrusion in the network. There are two phases in SA-NBC technique. The primary phase performs optimal feature selection and secondary executes attack detection and classification to improve the anomaly intrusion detection in wireless ad-hoc network. Initially, the feature of node is extracted through the simulated annealing. After that, the attack is identified and classified using NB classifier by considering selected optimal features. The brief explanation about the SA-NBC technique is presented in next subsections.

2.3.1 Simulated Annealing Method

In SA-NBC technique, the primary phase is used to perform the optimal feature selection with the aid of simulated annealing method. Generally, simulated annealing is a method used to solve the optimization issues.The global optimum feature from the number of features in the node is identified with the aid of SA method. After that, an efficient attack detection and classification is processed to increase the intrusion detection accuracy. In addition, SA is a probabilistic technique to detect the optimal results and avoiding the local optima when searching the solution space. In such a case, simulated annealing is an efficient technique in order to improve the network performance. The solid is efficiently heated to provide high temperature, then left to cool down slowly in simulated annealing process. During this process, a solid particle is travel into disordered state through heating the solid with higher temperature. This in turns, the internal energy gets improved. When gradually cool down, particles return to the order and it attains the equilibrium state at each temperature level. Lastly, the particle arrives a ground state at normal temperature and thus the internal energy minimized to a minimum level. Therefore, Simulated Annealing is applied in SA-NBC technique. The objective function value F with a simulated internal energy E, temperature T as control parameter, then frequent iteration as slowly decays of values thus provides the estimated optimal solution.

For allowing sufficient development, the initial temperature (T) is higher adequate. The temperature reduction function using SA is defined and it is formulated as follows,

$$T \leftarrow \gamma \times T. \tag{1}$$

In Equation (1), T denotes temperature it is based on the current temperature multiplied with the constant factor (γ). Initial solution is randomly chosen and it is taken as an optimal solution in SA. Next, the energy of the initial solution is calculated. A neighboring node of the initial state is chosen to calculate the energy when temperature T does not satisfies the termination condition.

The current state is replaced with a recently chosen state when the energy of newly selected neighboring node is less than or equal to the current state.If the energy of the new state is higher than the current state, a random value R is selected within the range of (0, 1). When random value R is less than the transition probability of the state, the optimal solution is achieved with the aid of simulated annealing method. After the temperature is decreased by using Equation (1), this process is continued until the termination condition T satisfied.

Consider the initial state of the node is'x' and the new state of the node is 'y', then the energy of the node is E(x) and E(y), the state transition probability is formulated as

$$P_{xy} = \exp(-\frac{E(y) - E(x)}{KT}). \tag{2}$$

From Equation (2), K represents the Boltzmann constant and T denotes the temperature of the material. The objective function value F is applied to describe the SA based feature selection algorithm for providing optimal solution. SA is a cooling scheme for finding optimal solutions to avoid local optima while searching the solution space. Based on the state transition using simulated annealing in SA-NBC technique, the optimal energy of the node is selected for classifying the network intrusion.

Algorithm 1 explains the process of finding an optimal state of the node with the aid of simulated annealing. The state transition carried out through the optimal state transition in the network. The energy of the new state is lesser than the current state and the current state is modified into new one. If the R value is less than the state transition probability, then the random number is generated. Hence, optimal feature of the node is chosen from the entire features. Then the intrusion detection and classification is processed with an optimal feature using NB classifier to detect the normal and anomalous node behavior in wireless ad-hoc network.

Algorithm 1 Simulated annealing algorithm

Input: Initial temperature (T), Number of nodes? $N_i = N_1, N_2, \cdots, N_n$, constant γ, a random number R
Output: Detect the optimal features for intrusion detection
for each node **do**
  Generate new state to choose optimal feature
  while Temperature > 0.001 **do**
    begin
    arbitrarily choose neighboring node state
    if $E(y) < E(x)$ **then**
      transition probability is calculated
    else
      produce R uniformly in the range (0, 1)
      if R < State transition probability **then**
        $x \leftarrow y$
        $T \leftarrow \gamma \times T$
      end if
    end if
  end while
end for

2.3.2 Naive Bayes Classifier For Anomaly Intrusion Detection

After selecting the optimal feature about the node, then the attacks detection and classification process is carried out using Naive Bayes Classifier. A Bayesian classifier is works on the design of a role of (natural) class that predict the values of features for nodes in the class. The nodes are grouped in classes since they include common values for the features. Such classes are further called natural kinds. If an organizer knows the class, the other features value of a node is predicted. If it unable to know the class, Bayes rule can be applied to predict the given class feature values of a node. The learning agent constructs a probabilistic model for classifying the node features whether it is normal or anomalous using Bayesian classifier. It also uses to predict the classification of a new pattern.In a probabilistic model, the classification is a latent variable in which the variable is probabilistically related to the detected variables.

Naive Bayes is a conditional probability model. The Naive Bayes classifier performs on optimal feature selection for the node. This means that the probability of one node feature does not affect the probability of the other. The independence of the Naive Bayesian classifier is realized in a certain trust network where the features are the nodes.

The nodes in the network is classified and represented by a vector $X = (x_1, x_2, \cdots, x_n)$. It denotes some optimal features of a node and assigns to this instance probabilities $p(C_{k|x_1}, x_2, \cdots, x_n)$ for each k possible classes C_k. For each example, the prediction can be computed by conditioning on detected values for the input features and by querying the classification.

In order to find the large number of optimal feature nodes n, the conditional probability can be expressed as,

$$p(C_{k|X}) = \frac{p(C_k)p(X|C_k)}{p(X)} \tag{3}$$

From Equation (3), $p(C_{k|X})$ denotes the posterior probability of class given predictor (feature), $p(C_k)$ represents prior probability of class features and $p(X|C_k)$ is the likelihood which is the probability of predictor given class. Then the $p(X)$ is a prior probability of features.

If the denominator is a normalizing constant then the probabilities provides the value greater than 1 means the node is normal. If else, the probabilities provides the value lesser than 1 means node is anomalous. Therefore, the intrusion detection accuracy is improved by using Naive Bayes classifier while monitoring the nodes behavior in wireless ad-hoc network.

The algorithmic process of NB classifier for anomaly intrusion detection is explained as follows,

Algorithm 2 Naive Bayes classifier algorithm

Input: Number of nodes $N_i = N_1, N_2, \cdots, N_n$
Output: normal and anomalous node classification
Begin
for each node **do**
  Evaluate t optimal feature nodes in vector X
  Measure conditional probability value using Equation (3)
  if $p(C_{k|X}) \geq 1$ **then**
    the node is normal
  else
    the node is anomalous
  end if
end for

Algorithm 2 shows the process of Naive Bayes classifier for detecting anomalous intrusion in wireless ad-hoc network. The vector representation is used to divide the nodes in different class. After that the conditional probability is applied to take optimal features of a node and classify the node is normal or anomalous. This in turns efficiently improves the anomaly intrusion detection accuracy. Therefore, Simulated Annealing based Naive Bayes Classifier (SA-NBC) technique is used to detect the intrusions and monitoring the network activities in an efficient manner.

3 Simulation Settings

In order to detect anomaly intrusion detection in wireless ad-hoc network, Simulated Annealing based NB Classifier (SA-NBC) technique is proposed and simulated using NS2 network simulator. The KDD cup 1999 dataset is taken from UCI repository for performing the simulation. KDD cup 1999 dataset contains standard set of data is audited, which comprises a number of intrusions in a network environment. The features are duration, src_bytes, dst_bytes,

number of urgent packets, srv_count, diff_srv_count and so on. Based on these features, the connection is separated in strong or feasible.

In wireless ad-hoc network, the number of nodes 500 is randomly arranged in an area $1500m \times 1500m$. Then speed of the node generates a traffic is maintained at a specific level as 20 m/s. The mobile nodes are distributed using Random Way point model in an area for simulation. Data packets used in the ranges from 10 to 100. The simulation time is taken as 1500sec. In each scenario, totally 500 nodes are used to identify the node interference and intrusion in the network. Table 1 illustrates simulation parameters.

Table 1: Simulation parameter

Parameter	Value
Network range	$1500m \times 1500m$
Simulation time	1500 ms
Number of mobile nodes	50, 100, 150, 200, 250, 300, 350, 400, 450, 500
Number of Data Packets	10, 20, 30, 40, 50, 60, 70, 80, 90, 100
Data Packets Size	100 - 512 KB
Range of communication	30 m
Speed of node	0 - 20 m/s
Mobility model	Random Way Point
Traffic type	Constant bit rate
Number of runs	10

4 Results and Discussion

Simulated Annealing based Naive Bayes Classifier (SA-NBC) technique is evaluated with the existing A3ACKs [17] and EAACK [15]. The experimental evaluation is carried out with the different parameters such anomaly intrusion detection accuracy, execution time and throughput. Performance is evaluated along with the following metrics with the help of tables and graph values.

4.1 Impact Of Anomaly Intrusion Detection Accuracy

The anomaly intrusion detection accuracy is measured as the ratio of the number of node accurately detected as anomalous to the total number of nodes in network. The anomaly intrusion detection accuracy is mathematical formulated as follows

$$AIDA = \frac{\text{No. of node accurately detected as anomalous}}{\text{No. of nodes}} \times 100. \quad (4)$$

From Equation (4), anomaly intrusion detection accuracy (AIDA) is measured in terms of percentage (%). If the anomaly intrusion detection accuracy is higher, then the method is said to be more efficient

Table 2: Tabulation for Anomaly intrusion detection accuracy

No. of nodes	Anomaly intrusion detection accuracy (%)		
	SA-NBC	A3ACKs	EAACK
50	84.20	75.24	70.22
100	86.54	78.16	72.35
150	87.16	80.30	74.55
200	89.52	82.54	75.63
250	90.41	85.28	77.22
300	92.10	86.86	79.46
350	93.47	89.45	81.42
400	95.32	91.36	83.53
450	97.27	93.44	86.40
500	98.14	94.27	89.18

Table 2 shows the tabulation for anomaly intrusion detection accuracy using proposed SA-NBC compared with existing A3ACKs [17] and EAACK [15] methods in wireless ad-hoc network. Number of nodes is taken from the range of 50 to 500 for experimental purpose. From table, it is clear that anomaly intrusion detection accuracy is increased for the respective increase in number of nodes using all the methods.

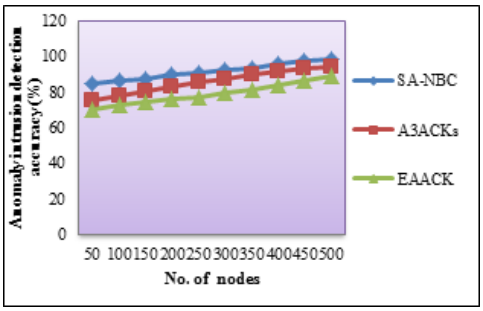


Figure 2: Measure of anomaly intrusion detection accuracy

Figure 2 shows the performance of anomaly intrusion detection accuracy using three methods such as proposed SA-NBC and existing A3ACKs [17] and EAACK [15] methods. From the figure, it is clearly illustrated that the anomaly intrusion detection accuracy is improved in SA-NBC technique. This efficient improvement in SA-NBC technique is attained with the help of simulated annealing based classification in wireless ad-hoc network. Then the Naive Bayes classifier is used to classify the normal and anomalous nodes in the network based on the optimal feature selection. The condition probability values are considered to detect the anomalous nodes accurately. There-

fore, the anomaly intrusion nodes are detected in wireless ad-hoc network and thus improve accuracy of anomaly intrusion detection using SA-NBC technique by 6% and 14% compared to existing A3ACKs [17] and EAACK [15] methods respectively.

4.2 Impact of Execution Time

Execution time is measured by product of time taken for detecting an intrusion or attacks in a network with respect to number of nodes participate in that network. It is formulated as given below.

$$ET = n \times \text{Time (intrusion detection)}. \quad (5)$$

From Equation (5), Execution Time 'ET' is measured in terms of milliseconds (ms). Lower execution time ensures the effectiveness of method.

Table 3: Tabulation for Execution time

No. of packets	Execution time (ms)		
	SA-NBC	A3ACKs	EAACK
50	9.53	11.21	13.76
100	13.59	14.68	17.28
150	17.24	18.57	20.63
200	19.48	20.65	22.69
250	22.47	23.58	26.49
300	25.38	27.49	29.66
350	28.32	29.37	32.52
400	30.26	32.56	34.25
450	33.26	35.26	37.24
500	35.12	37.31	39.15

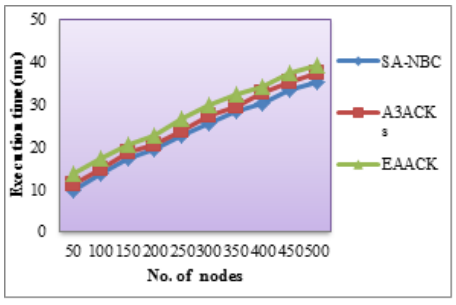


Figure 3: Measure of execution time

Table 3 and Figure 3 show the measure of execution time with respect to varying number of nodes in proposed SA-NBC technique compared with existing A3ACKs [17] and EAACK [15] methods. As shown in Figure 4, the proposed SA-NBC technique provides better reduction in execution time for intrusion detection when compared to other existing methods. This efficient reduction of execution time is achieved by the application of Naive Bayes classifier for MANETs. Data packets are transmitted through network from source node to destination node and entire information successfully transferred using proposed SA-NBC technique due to the detection of abnormal node in the network. Hence delay time for transmitting data packets to destination node is reduced effectively. Therefore, execution time for intrusion detection using proposed SA-NBC technique is reduced by 8% when compared to A3ACKs [17] and 19% when compared to EAACK [15] method respectively.

4.3 Impact of Throughput

Throughput is measured by the ratio of successfully received data packets at destination node and total number of data packets sent through source node. Throughput rate in wireless ad-hoc network is calculated as shown below.

$$T = \frac{\text{successfully received data packets}}{\text{total number of data packets sent}} \times 100. \quad (6)$$

From Equation (6), throughput 'T' is measured in terms of percentage (%). If throughput rate is high, then the network is said to be more secure and efficient.

Table 4: Tabulation for throughput

No. of packet sent	Throughput (%)		
	SA-NBC	A3ACKs	EAACK
10	77	70	62
20	79	72	64
30	82	75	66
40	84	77	67
50	85	78	69
60	88	80	71
70	91	83	74
80	93	85	76
90	95	86	78
100	96	87	81

Table 4 shows the measure of throughput using proposed SA-NBC technique compared with existing A3ACKs [17] and EAACK [15] methods. The number of data packets is taken as the ranges from 10 to 100 for the experimental purpose. From the table, the throughput of the network is increased with the respective increase in data packets. Proposed SA-NBC technique provides higher throughput when compared to state-of-the-art methods.

Figure 4 depicts the result analysis of the throughput with number of data packets are considered for evaluation process. From the figure, it is clearly evident that the proposed SA-NBC technique accurately classifies the node as normal or anomalous. This technique improves the performance results when compared to existing methods due to NB classifier in SA-NBC technique. The NB

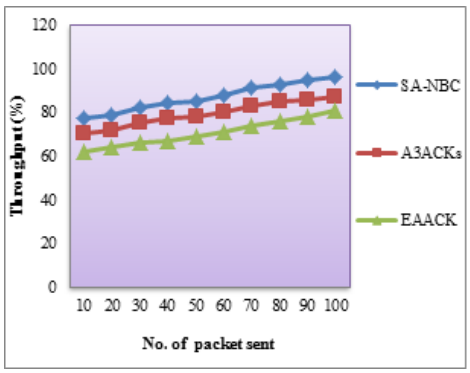


Figure 4: Measure of throughput

model classifies the nodes as normal or anomalous in vector consideration in which the nodes are selected with optimal features. With the aid of selected optimal features of a node, the traffic pattern is analyzed in the network. In addition, simulated annealing is used to perform the optimal feature selection process for further classify the node is normal or anomalous in the wireless ad-hoc network. Hence, efficient communication is achieved while transmitting the data packets through the normal node and thus improves the performance of the network. This in turns efficient throughput is achieved in wireless ad-hoc network. Therefore, proposed SA-NBC technique improves the throughput by 9% and 19% compared to existing [17] and EAACK [15] respectively.

5 Related Works

The security attacks and intrusion detection systems method was introduced in [11] for self-configurable networks. However, the applications are used in the method was affected by intruders. A novel intrusion detection system based on the trust rates was introduced in [5] to detect the intrusive action in MANET. But, the execution time for intrusion detection was remained unaddressed.This issue is overcome by the SA-NBC technique for improving the performance through identifying the anomalous node behavior.

Distributed combined authentication and intrusion detection was designed in [2] to maximize security in MANET. But, trust values from all nodes were not combined effectively. A behavior-rule specification-based technique was introduced in [9] for intrusion detection in medical devices. However, the large numbers of nodes are does not handled effectively. The SA-NBC technique improves the optimal feature for classifying large number the anomalous and normal activities in wireless ad-hoc network.

An intelligent multi-level classification technique was introduced in [3] to detect the intrusion detection in MANET. The mixture of tree classifier with labeled training data and enhanced multiclass classifier algorithm was designed to prevent the network from the intrusion. But, the intrusion detection was does not efficiently carried out. This problem is solved by SA-NBC technique by using Naive Bayes classifier. Neural network method was introduced in [14] to distinguish the normal and attacked behavior of the system based on MLP. But, the normal and anomalous of the system was not distinguished.

TermID, a distributed rulebased network intrusion detection system was developed in [6] for performing intrusion detection applications in wireless networks. But, the classification was not enhanced using distribution of the tasks in wireless networks.The SA-NBC technique used to improve the intrusion detection effectively by means of selecting optimal features of a node in the network. In [7], IDS based on self-learning technique was developed to detect the attacks in the network where the system uses unknown data pattern classifier (Neuro-fuzzy approach) thus reduces the dimensionality of the dataset. However, the classification was not efficient.

A novel IDS technique of cluster leader election process and a hybrid IDS was introduced in [18]. It provides the intrusion detection service by means of Vickrey Clarke-Groves mechanism in MANET. However, the intrusion detection rate was not enhanced and reduces the false positive rate. The cross-layer based distributed machine learning anomaly detection system was developed in [**?**] to protect the system. However, the throughput was not improved. This problem is addressed and it reduced in SA-NBC technique using Simulated annealing based Naive Bayes classifier.

6 Conclusion

A novel technique is called Simulated Annealing based Naive Bayes classifier (SA-NBC) is proposed to detect Anomaly Intrusion Detection in wireless ad-hoc network. An anomaly-based intrusion detection system is an essential one to observe the network activities and classify whether it either normal or anomalous node. Hence, the accuracy of intrusion detection is enhanced. In proposed SA-NBC technique, simulated annealing is used to choose the optimal feature of the node and thus detect the intrusion in the network. Based on these optimal features, the Naive Bayesclassifier is used to classify the malicious node and normal node with the aid of calculating conditional probability. It outlines the vector representation for detecting the network intrusions and observes network behavior and classifying the node as either normal or abnormal (anomalous). The experiments are conducted on different parameters such as anomaly intrusion detection accuracy, execution time and throughput. The performance results show that the proposed SA-NBC technique improves the anomaly intrusion detection accuracy, throughput and reduces the execution time than the state-of-art methods.

References

[1] V. N. T. AlkaChaudhary and A. Kumar, "A new intrusion detection system based on soft computing techniques using neuro-fuzzy classifier for packet dropping attack in MANETs," *International Journal of Network Security*, vol. 18, no. 3, pp. 514–522, May 2016.

[2] S. Bu, F. R. Yu, X. P. Liu, P. Mason, and H. Tang, "Distributed combined authentication and intrusion detection with data fusion in high-security mobile ad hoc networks," *IEEE Transactions on Vehicular Technology*, vol. 60, no. 3, pp. 1025–1036, Mar. 2011.

[3] S. Ganapathy, P. Yogesh and A. Kannan, "An intelligent intrusion detection system for mobile ad-hoc networks using classification techniques," *Advances in Power Electronics and Instrumentation Engineering*, vol. 148, pp. 117–122, 2011.

[4] W. R. Ghanem, M. Shokair and M. I. Dessouky, "Defense against selfish PUEA in cognitive radio network based on hash message authentication code," *International Journal of Electronics and Information Engineering*, vol. 4, no. 1, pp. 12–21, 2016.

[5] D. G. Gopal1 and R. Saravanan, "A novel trust based system to detect the intrusive behavior in MANET," *International Journal of Electronics and Information Engineering*, vol. 3, no. 1, pp. 31–43, 2015.

[6] C. Kolias, V. Kolias, G. Kambouraki, "TermID: A distributed swarm intelligence-based approach for wireless intrusion detection," *International Journal of Information Security*, vol. 16, no. 4, pp. 401–416, 2017.

[7] B. Mahapatra, S. Patnaik, "Self adaptive intrusion detection technique using data mining concept in an ad-hoc network," *Procedia Computer Science*, vol. 92, pp. 292–297, 2016.

[8] S. Mamatha and A. Damodaram, "Intrusion detection system for mobile ad hoc networks based on the behavior of nodes," *International Journal of Grid Distribution Computing*, vol. 7, no. 6, pp. 241–256, 2016.

[9] R. Mitchell and I. R. Chen, "Behavior rule specification-based intrusion detection for safety critical medical cyber physical systems," *IEEE Transactions on Dependable and Secure Computing*, vol. 12, no. 1, pp. 16–30, 2015.

[10] A. Mitrokotsa and C. Dimitrakakis, "Intrusion detection in MANET using classification algorithms: The effects of cost and model selection," *Ad Hoc Networks*, vol. 11, pp. 226–237, 2013.

[11] N. Mohd, S. Annapurna, H. S. Bhadauria, "Taxonomy on security attacks on self configurable networks," *International Journal of Electronics and Information Engineering*, vol. 3, no. 1, pp. 44–52, 2015.

[12] F. Nabi, M. M. Nabi, "A process of security assurance properties unification for application logic," *International Journal of Electronics and Information Engineering*, vol. 6, no. 1, pp. 40–48, 2017.

[13] S. OzgeCepheli G. Kurt, "Hybrid Intrusion detection system for DDoS attacks," *Journal of Electrical and Computer Engineering*, vol. 2016, pp. 1–8, 2016.

[14] K. Pavani and A. Damodaram,. "Multi-class intrusion detection system for MANETs," *Journal of Advances in Computer Networks*, vol. 3, no. 2, pp. 93–98, 2015.

[15] E. M. Shakshuki, N. Kang and T. R. Sheltami, "EAACK - A secure intrusion-detection system for MANETs," *IEEE Transactions on Industrial Electronics*, vol. 60, no. 3, pp. 1089–1098, 2013.

[16] N. Shah and S. Valiveti, "Intrusion detection systems for the availability attacks in ad-hoc networks," *International Journal of Electronics and Computer Science Engineering*, vol. 1, no. 3, pp. 1850–1857, 2012.

[17] T. Sheltami, A. Basabaa and E. Shakshuki, "A3ACKs: Adaptive three acknowledgments intrusion detection system for MANETs," *Journal of Ambient Intelligence and Humanized Computing*, vol. 5, no. 4, pp. 611-620, 2014.

[18] B. Subba, S. Biswas, S. Karmakar, "Intrusion detection in mobile ad-hoc networks: Bayesian game formulation," *Engineering Science and Technology*, vol. 19, pp. 782–799, 2016.

[19] M. Usha and P. Kavitha, "Anomaly based intrusion detection for 802.11 networks with optimal features using SVM classifier," *Wireless Networks*, vol. 23, no. 8, pp. 2431–2446, 2017.

Biography

K. Murugan is Assistant Professor and Head of the Department of Computer Science at Government College for Women, Kolar. His current area of research interest is Computer Networks and its applications. He has successfully guided 15 candidates for M.Phil. He has been teaching computer Science for the past 18 Years. He completed his Master Degree in Computer Science at Bharathidasan University, Master of Philosophy in Computer Science at Manonmaniam Sundaranar University, Master of Engineering in Computer Science and Engineering at Anna University.

P. Suresh is Associate Professor and Head, Department of Computer Science, Salem Sowdeswari College [Govt. Aided], Salem. He received the M.Sc, Degree from Bharathidasan University in 1995, M.Phil Degree from Manonmaniam Sundaranar University in 2003. The M.S (By Research) Degree from Anna University, Chennai 2008 in Science and Humanities. PGDHE Diploma in Higher Education and Ph.D Degree from Vinayaka Missions University in 2010 and 2011 respectively in Computer Science. He is an Editorial Advisory Board Member of Elixir Journal. His research interest includes Data Mining and Natural Language Processing. He is a member of Computer Science Teachers Association, New York.

Identification of Cyber Criminal by Analyzing Users Profile

K. Veena and K. Meena
(Corresponding author: K. Veena)
Department of Computer Science and Engineering, VelTech Dr. RR and Dr. SR University
400 Feet, Outer Ring Road, Avadi, Chennai 600 062, India
(Email: veenakanagaraj07@gmail.com)
(Received Mar. 27, 2017; revised and accepted July 29, 2017)

Abstract

This paper presents a method to analyze the feedback from various users and thus determine the cyber criminal. The cyber criminals are effectively identified by applying the various clustering techniques for different number of users with different attributes (characters). The purpose of clustering is to identify natural groupings of data from a large data set to represent the system's behavior. The clustering is done using the various techniques like the Gaussian technique, K Means Clustering, Fuzzy C Means Clustering and Fuzzy Clustering. Clustering of numerical data forms the basis of many classification and system modeling algorithms. The data that true without any false information is take as the called the genuine data and the data that contains false information is taken the crime data. By clustering, the genuine data (Cluster 0) is eliminated and only the crime data (Cluster 1) is taken. From the genuine data the false positive is taken as the crime data. From the criminal data the true negative is also eliminated. The criminal data is further analyzed using the various classes and then the criminal is detected. Many of the researchers used minimum number of attributes to identify the criminal. In order to increase the crime identification rate, this paper uses 25 various attributes which are collected from 25 users in different scenarios. In this paper, the profile of the person involved in cyber crime is analyzed for further calculations. By identification of the profile of the cyber criminal, the detection of the crime can be done. In this paper 25 users along with 25 attributes is taken as experimental investigation.

Keywords: Cyber Criminal; Fuzzy Clustering; Identification

1 Introduction

A computer and a network can be used to commit a crime refers to as Cyber Crime. The meeting on Cyber crime was the first international treaty which was conducted to understand the Computer crime and Internet crimes. The convention on cyber crime was the first international treaty that seek to address the computer crime and internet crimes by harmonizing the national laws [7], thus improving the investigative techniques [17] and increasing the cooperation among nations.

Cyber crime is a world wide criminal phenomenon which confuses the customary distinction between fear to criminal and terrorist activity i.e internal and military i.e external security and does not respond to single authority approaches to policing.The liability of networks to exploitation for a number of different ends, and the ease with which individuals may move from one type of illegal activity to another suggests that territorialism in all its forms (both of nations and regions, and specific authorities within nations) hinders efforts to successfully combat the misuse of communications technology. There has been a rise to the industrialization of a type of crime where the commodity, personal information, moves far too quickly for conventional law enforcement methods to keep pace. Stolen personal and financial data - used, for example, to gain access to existing bank accounts and credit cards, or to fraudulently establish new lines of credit - has a monetary value [8, 11].

Whenever a cyber crime is committed the victim suffers silently. He/she is not able to speak openly and accept that he/she is a victim of cyber crime. If the victim is an Indian her case is more ridiculous. She is blamed first, hence they do not express their difficulty outside. In this paper my aim is to help such victims who suffer silently. They should just give a complaint and the set of suspects. The criminal has to be detected [2].

In the paper, GPS Spoofing Detection Based on Decision Fusion with a K-out-of-N Rule [18] to get a high detection probability of the GPS spoofing, decision fusion is proposed and three classifiers are used and the results are fused with K-out-of-N decision rule and the final classification is obtained.

In the paper, Sheu, "Distinguishing Medical Web

Pages from Pornographic Ones: An Efficient Pornography Websites Filtering Method" [15], the uncomplicated decision tree data mining algorithm is used to determine the association rules about the pornographic and medical web pages.

In the paper, "Clustering based K-anonymity algorithm for Privacy preservation" [9] K-anonymity is used as a effective model for protecting privacy while publishing data. A clustering based K-anonymity algorithm is used and it is optimized with parallelization.

In the paper, "A Quantitative and Qualitative Analysis-based Security Risk Assessment for Multimedia Social Networks", much attention is given to the spreading and sharing of personal information in the social media. Social media can be used to follow a person [22].

Cyber crime is a truly global criminal phenomenon which blurs the traditional distinction between threats to internal (criminality and terrorist activity) and external (*i.e.* military) security and does not respond to single jurisdiction approaches to policing.

Here the various clustering techniques are applied for different number of users with different attributes (characters). The input data is further analyzed for different threshold values. The profile of the person involved in cyber crime is analyzed for further calculations [19] by determining the cluster formed using the various clustering techniques. The profile of the cyber criminal is identified. The psychology of cyber criminology directs its attention towards the application of the physical, psychological, social relationships and mental characteristics, as well as towards the evidence of the cybercrime [5,10,20,21]. The computer may have been used in the commission of a crime, or it may be the target, or the user of the computer might have been the target.

The present paper presents a method to analyse the feedback from various users and thus determine the cyber criminal. By clustering, the genuine data (Cluster 0) is eliminated and only the crime data (Cluster 1) is taken [21]. From the genuine data the false positive is also taken as the crime data. From the Crime data the true negative is taken as Genuine data and added to the Cluster 0. The criminal data is further analyzed [14] using the various classes as Class as None, Soft and Hard and then the criminal is detected.

The various clustering techniques are applied for different number of users with different attributes (characters). The data is further analyzed for different threshold values [1]. The profile of the person involved in cyber crime is analyzed for further calculations.

The paper is organized as follows. In Section 1, the abstract is given and a little review of the entire paper. It is followed by the introduction which gives the necessity of detection of cyber crime. Then the motivation and contribution of the proposed work is given. Then the justification of clustering methods are given which consists of the algorithm used.

In Section 2, the methodology used is given which consists of different users, attributes and threshold value, the various clusters identified. Then the comparative results and the results of various clustering techniques are given.

In Section 3, the various classification of clustering is given such as Cluster and Fuzzy C-Means (FCM), Formation of Clusters using the Gaussian Mixture Models. The reasons for choosing Gaussian Clustering Technique is also given.

In Section 4, the implementation and the results are given. The identification of the criminal is also given. A brief description of the Gaussian Clustering Analysis, K Means Clustering Analysis, Fuzzy C Means Clustering Analysis and Fuzzy Clustering Analysis with various Users and Attributes [4, 6, 12, 13, 16]. are given. Finally, the conclusion, acknowledgement and references are given.

2 Justification of Clustering Method

2.1 The Proposed Method

The various clustering methods used here are the Gaussian technique, K Means Clustering, Fuzzy C Means Clustering and Fuzzy Clustering. The clusters are formed based on these clustering techniques. The users profile which consists of attribute set 1 with 25 users are analyzed with the analyzer 1. After the determination of the clusters the clusters are classified as Cluster 0 and Cluster 1. The false positive data is removed in the cluster 0 and the true negative is removed in the cluster 1. The data is further classified as Soft and Alert, Hard and Criminal and None and Genuine based on the average rate as 4-6, 7-9 and 0-3. After classification if the criminal cannot be determined then it is further checked with Analyzer 2, which consists of another set of 15 attributes.

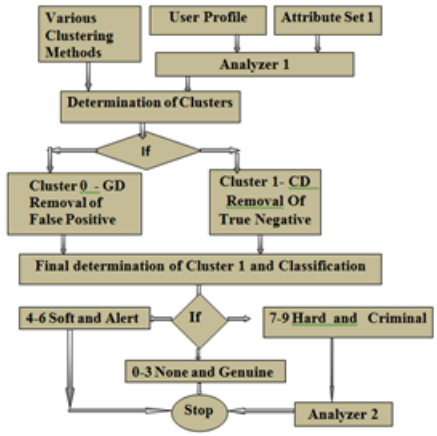


Figure 1: Cyber crime detection flowchart

2.2 Algorithm Used

To analyse the information from various users, the term entropy is the measure of disorder or user data in normal state. It contains the positive and negative values of the cluster formed.

$$\text{Entrophy}(S) = -p_{(+)} \log_2 p_{(+)} - p_{(-)} \log_2 p_{(-)} \quad (1)$$

Algorithm 1 Determination of the criminal

```
Begin
Initialize User Profile and Attribute Set 1
Input data to Analyzer 1, go to Step 5
Sent request to Analyze Data using various clustering methods
Determine the clusters as Genuine data(0) and Crime data(1)
if Cluster 0 then
    print as Genuine data and remove False Positive
    Goto Step 8
else
    print as Crime data and remove True Negative
end if
Final determination of Cluster 1 and Classification
if result is in range 0-3 then
    print as None and Genuine Data
    goto Step 13
end if
if result is in range 4-6 then
    print as Soft and Alert Data
    goto Step 13
else
    if result is in range 7-9 then
        print as Hard and Crime Data
    end if
end if
Goto Analyzer 2
Stop
```

3 Methodology

3.1 Optimization of Attributes Used

Here the data (`http//www.kdnuggets.com/datasets`) taken is for 25 user with various with 25 different attributes (attributes set 1). The various types of Cybercrime which are used as attributes are, given in Table 1. The percentage of the attributes differs in different regions, which is indicated in the table. The attributes set 2 is taken if the criminal cannot be determined with attribute set 1. The attribute set 2 consists of the data consecutive four years before the crime was committed, three months before, three days before the crime, three days after the crime, the day the crime was committed and the relation ship between the criminal and the victim.

Analysis of various attributes with their locations in percentage Table 1.

3.2 Method Used

The data is taken for different number of users, with various attributes and different threshold values. The clusters are formed based on the cut off values. If the cluster falls below the threshold value, the cluster is in "0", otherwise the cluster is in "1". The Cluster 0 is taken as the "Genuine Data" and the Cluster 1 is taken as the "Crime data".

The Different Users, Attributes and Threshold Value Table 2.

3.3 Classification of Clustering

Cluster is a group of objects that belongs to the same class. In other words, similar objects are grouped in one cluster (legal) and dissimilar objects (illegal) are grouped in another cluster. A cluster of data objects can be treated as one group. While doing cluster analysis, we first partition the set of data into groups based on data similarity and then assign the labels to the groups. The main advantage of clustering over classification is that, it is adaptable to changes and helps single out useful features that distinguish different groups. Clustering also helps in classifying documents on the web for information discovery. Clustering is also used in outlier detection applications such as detection of credit card fraud [3]. As a data mining function, cluster analysis serves as a tool to gain insight into the distribution of data to observe characteristics of each cluster. The various Clustering Methods are Partitioning Method, Hierarchical Method, Density-based Method, Grid-Based Method, Model-Based Method, Constraint-based Method and Fuzzy C Means Clustering.

Clustering with Gaussian Mixtures

The Gaussian mixture distributions can be used for clustering data, by realizing that the multivariate normal components of the fitted model can represent the clusters. To demonstrate the process, first some simulated data is generated from a mixture of two bivariate Gaussian distributions using the **mvnrnd** function: The probability density function of the d-dimensional multivariate normal distribution is given by the formula, where

$$y = f(x, \mu, \Sigma) = \frac{1}{\sqrt{|\Sigma|(2\pi)^d}} e^{-\frac{1}{2}(x-\mu)\Sigma^{-1}(x-\mu)'}$$

where x and μ are 1-by-d vectors and Σ is a d-by-d symmetric positive definite matrix. Only random vector generation is supported for the singular case.

Partition into Clusters

Then fit the two-component Gaussian mixture distribution. Here the correct number of components

Table 1: Analysis of various attributes with their locations in percentage

Attributes set1	**Browsing Centre**	**Institution**	**Household**	**Mobile**	Medical Shop
Hacking	100	100	100	100	100
Theft	100	100	100	100	100
Cyber Stalking	100	100	100	100	100
Identity theft	100	100	100	100	100
Malicious Software	85	85	85	85	85
Child Soliciting	100	100	50	100	50
Child Abuse	100	100	100	100	100
Assault by Threat	100	100	100	100	100
Child Pornography	100	100	100	100	100
Cyber Illegal imports	85	85	85	85	85
Cyber Laundering	100	100	50	85	85
Cyber Terrorism	100	100	50	85	85
Cybertheft	100	100	50	85	85
Advertising	25	25	25	25	25
Soliciting harlotry	100	100	100	100	100
Drug Sales	25	100	85	25	25
Frequency	25	25	25	25	25
Malicious Code	50	50	25	50	50
Password Violations	85	85	25	85	85
Excess Privileges	50	85	25	50	50
Data Forwarding	25	25	25	25	25
Computer related offences	100	85	85	85	85
Publication irrelevant content	100	85	50	50	50
Transmission of obscene conten	100	100	100	100	100
Sexually explicit content	100	100	100	100	100

Table 2: The different users, attributes and threshold value

Properties	**Data Set1**	**Data Set2**	**Data Set3**	**Remarks**
Users	25	15	5	The number of users are decreased
Attributes	25	15	7	The number of attributes are decreased
Threshold Value	5	3	3	Threshold Value is gradually decreased
Number of Cluster 0	11	3	1	Users and Clusters are propotional
Number of Cluster 1	14	12	4	Users and Clusters are propotional
Remarks	1Greater	1 Greater	1 Greater	

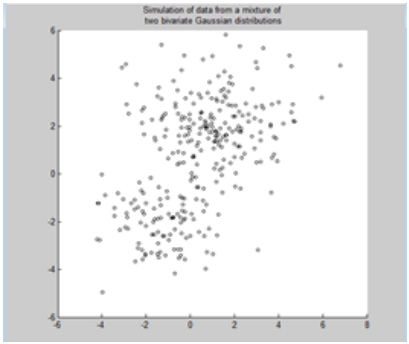


Figure 2: Simulation of data from a mixture of two bivarieties Gaussian distribution

Figure 3: Partition into clusters

is used is two. This data displays 28 iterations, log-likelihood = 1223.66.

Then plot the estimated probability density contours for the two-component mixture distribution. The two bivariate normal components overlap, but their peaks are distinct. From this data it can be concluded that the data could be divided into two clusters. Partition the data into clusters using the cluster method for the fitted mixture distribution. The cluster method assigns each point to one of the two components in the mixture distribution.

3.4 Reasons For Choosing Gaussian Clustering Technique

Analysing the various clustering techniques, it is clear that the Gaussian Clustering Technique is useful compared to the other techniques. In the Gaussian Clustering with various Users and Various Attributes, the number of Cluster 0 and Cluster 1 obtained is different. The iteration Count and the log-like hood remains the same if the users is equal to the number of attributes. The iteration count and the log-like hood differs if the users if different from the attributes. For different Users and attributes with the same threshold value, the iteration count and the log-like hood is different.

4 Implementation and Results

Here various clustering techniques like Gaussian Clustering, K Means Clustering, Fuzzy Means Clustering and Fuzzy Clustering are taken with different data sets. The iteration count decreases in Gaussian Clustering, remains the same in K Means Clustering, increases in Fuzzy C Means Clustering and reduces drastically for Fuzzy Clustering.

Performance Evidence of Various Clustering Techniques Table 3.

In the Gaussian Clustering with various Users and Various Attributes, the number of Cluster 0 and Cluster 1 obtained is different. The iteration Count and the log-like hood remains the same if the users is equal to the number of attributes. The iteration count and the log-like hood differs if the users if different from the attributes. For different Users and attributes with the same threshold value, the iteration count and the log-like hood is different. In the K Means Clustering Analysis with various Users and Attributes, the iteration count is the same and it is 5. But the iteration occurs at different values. The random data obtained also differs. In Fuzzy C Means Clustering the iteration is different for Users. If the number of Users are less then the number of Iteration Count increases. The Object Function is the same, if the threshold value is the same. In Fuzzy Clustering for different users there is different iteration count and different object function. The performance and the time taken for the various data set is shown in table IV. The training used is Scaled Conjugate Gradient, the performance used is Mean Squared Error and the Data Division used is Random.

Neural Network Training for various data set Table 4.

The experimental analysis is done with 25 users and each user has 25 attributes. The various users $1, 2, \cdots, 25$. For every user 25 attributes are taken and numbered as $1, 2, \cdots, 25$. The sum of all the attributes for each user is calculated. The average value of the attributes for each user is calculated. If the average value o is greater than 4 then it is cluster 1 called the (crime data) or o if it is less than 4 then it is cluster 0 called the (genuine data) The number of Cluster 0 is 11 and the various users are 3, 4, 5, 6, 7, 10, 11, 16, 19, 20, 25 and the number of Cluster 1 is 14 and the various users are 1, 2, 8, 9, 12, 13, 14, 15, 17, 18, 21, 22, 23, 24. The number of attribute with value 1 is 51, the number of attribute with value 2 is 194, the number of attribute with value 3 is 452, the number of attribute with value 4 is 132, the number of attribute with value 5 is 320, the number of attribute with value 6 is 146, the number of attribute with value 7 is 75, the number of attribute with value 8 is 94, the number of attribute with value 9 is 60 and the number of attribute with value 10 is 0.

Since the highest is number 3 with value 452 and the lowest is number 1 with value 51,to determine the false positive, the reference is taken as attribute 3 and attribute 1. The number of value 1 and value 3 is counted in each user and then the average is taken. In Cluster 1,

Table 3: Performance evidence of various clustering techniques

Set	**Gaussian**	**K Means**	**Fuzzy C**	**Fuzzy**
Data Set	Count and Log	Count and Data	Count and Object	Count and Object
Data Set1	28 and -1223.66	5(4,5,6,7,8) and 302.923	41 and 787.28	50 and 1180.92
Data Set2	28 and-1223.66	5(3,6,7,8,10) and 295.87	46 and 301.29	100 and 451.93
Data Set3	26 and -1215.09	5(4,4,5,6,6) and 310.605	96 and 301.29	22 and 42.34

Table 4: Neural network training for various data set

Progress	**Epoch**	**Time**	**Performance**	**Gradient**	**Validation Checks**	**Best Validation Performance**
Data Set1	8	0:00:01	19.9/41.0	1.84	6	21.3396 at Epoch 2
Data Set2	8	0:00:01	23.5/40.7	2.15	6	33.4325 at Epoch 2
Data Set3	25	0:00:08	0.000872/0.384	0.000304	6	0.025708 at Epoch 19

the genuine data is to be eliminated. The least number is taken as Genuine data *i.e.*, false positive and added to the Genuine data. It is added to cluster 0. The genuine users are user 1, user 9 and user 18.

In Cluster 0, the crime data is to be added. The highest number is taken as the Crime data *i.e.* true negative and added to the crime data. It is added to cluster 1. The crime data users are user 6, user 20 and user 25.

The user 1, user 9 and user 18 are Genuine data so add them to cluster 0, the user 6, user 20 and user 25 are criminal data so add them to cluster 1. The final users in the crime data are 2, 8, 12, 13, 14, 15, 17, 21, 22, 23, 24, 6, 20, 25. To determine the decision algorithm, the classification is done as if the sum of the range of attributes is from 0-3 it is none classification and the data is genuine, if the sum of the range of attributes is from 4-6 the classification is soft and the data is alert and if the sum of the range of attributes is from 7-9 the classification is hard and the data is criminal. The number of count in each criteria is taken and then the highest number is taken in each classification.

The Result is based on "If no cell is selected then it is NONE". Otherwise "it is either HARD or SOFT". The Result 1is based on "If all cell is selected it is HARD", "if any two cell is selected it also HARD". Otherwise "it is SOFT". The Result is based on if all three cells are selected it is HARD and that user is the criminal. If only two cell is selected, then it is HARD. Whether that user is the criminal, has to be analyzed further. If there is a tie with the users such as, more number of users are in the HARD classification, Then it is further classified with another 15 attributes, and then the criminal is detected. Table 5 shows the determination of the criminal.

From the above formed Cluster 1, the genuine data is removed, the average of Count 1 and Count 3 is taken. The least of the average is taken as the Genuine data. The User 1, User 9 and User 18 are with the least average and hence they are considered as the Genuine Data. The User 1, User 9 and User 18 are added to the Cluster 0. From the formed Cluster 0, the crime data is removed the sum of Count 1 and Count 3 is taken. The least of the Sum is taken as the Crime data. The User 6, User 20 and User 25 are with the highest sum and hence they are considered as the Crime Data. The User 6, User 20 and User 25 are added to the Cluster 1. The table shows the cluster of the Crime data which is used to determine the crime. The True Negative is removed and false positive is added. The data is further classified as Soft, Alert and Hard. Here the user 8 and User 17 are in the same a range and hence further analysis is to be done with the Attribute set 2. Since User 8 and User 17 are having the same features, they are further analyzed with another attributes set 2 and the criminal is detected. It is **User 8**.

5 Conclusions

Analysing the various clustering techniques, it is clear that the Gaussian Clustering Technique is useful compared to the other techniques. In the Gaussian Clustering with various Users and Various Attributes, the number of Cluster 0 and Cluster 1 obtained is different. The iteration Count and the log-like hood remains the same if the users is equal to the number of attributes. The iteration count and the log-like hood differs if the users if different from the attributes. For different Users and attributes with the same threshold value, the iteration count and the log-like hood is different.

1) The feedback from various users are analyzed and thus the cyber criminal is determined. The Clustering of numerical data was used as the basis of classification and system modeling algorithms. The purpose of clustering was to identify natural groupings of data from a large data set to produce a concise representation of a system's behavior. After clustering the genuine data (Cluster 0) is eliminated and only the

Table 5: Determination of the criminal

Users	**Average**	**Cluster**	**GD Removal**	**CD Add**	Classification	Result	Classification2	Result2
1	4.52	CD	GD					
2	4.28	CD			CD	HARD/SOFT	SOFT	
3	4	GD						
4	3.28	GD						
5	3.96	GD						
6	3.2	GD		CD	CD	HARD/SOFT	SOFT	
7	3.6	GD						
8	4.32	CD			CD	HARD/SOFT	HARD	CRIMINAL
9	4.76	CD	GD					
10	3.92	GD						
11	3.72	GD						
12	4.28	CD			CD	NONE		
13	4.12	CD			CD	HARD/SOFT	SOFT	
14	4.16	CD			CD	NONE		
15	4.56	CD			CD	HARD/SOFT	SOFT	
16	3.6	GD						
17	4.32	CD			CD	HARD/SOFT	HARD	
18	4.76	CD	GD					
19	3.92	GD						
20	3.72	GD		CD	CD	NONE		
21	4.28	CD			CD	NONE		
22	4.12	CD			CD	HARD/SOFT	SOFT	
23	4.16	CD			CD	NONE		
24	4.56	CD			CD	HARD/SOFT	SOFT	
25	3.72	GD		CD	GD	NONE		

crime data (Cluster 1) was taken. From the genuine data the false positive was also taken as the crime data. Before the criminal is detected, from the criminal data the true negative was also eliminated. The criminal data was further analyzed using the various classes and then the cyber criminal was detected.

2) The various clustering techniques was applied for different number of users with different attributes (characters). The data was further analyzed for different threshold values. The profile of the person involved in cyber crime was analyzed for further calculations.

3) The reasons for choosing each classification are given below. In the Gaussian Clustering with various Users and Various Attributes, the number of Cluster 0 and Cluster 1 obtained is different. The iteration Count and the log-like hood remains the same if the users is equal to the number of attributes. The iteration count and the log-like hood differs if the users if different from the attributes. For different Users and attributes with the same threshold value, the iteration count and the log-like hood is different. In the K Means Clustering Analysis with various Users and Attributes, the iteration count is the same and it is 5. But the iteration occurs at different values. The random data obtained also differs. In Fuzzy C Means Clustering the iteration is different for Users. If the number of Users are less then the number of Iteration Count increases. The Object Function is the same, if the threshold value is the same. In Fuzzy Clustering for different users there is different iteration count and different object function.

Acknowledgments

The cyber crime and security is a sensitive topic and many of the victims may not wish to speak about it openly. Hence as I would greatly like to help people in this regard using data mining techniques. I owe a great many thanks to a great many people who helped and supported me during the writing of this paper. The author wishes to express her gratefulness to the reviewers and for the chance to profit from the considerate and useful comments. The author wishes generally to emphasize that she is indebted to many ideas raised by other literature sources and is grateful for further and suggestions.

References

[1] O. M. A. Abbas, "Comparisons between data clustering algorithms," *International Arab Journal of Information Technology*, vol. 5, no. 3, pp. 320–325, July 2008.

[2] K. S. Arthisree and A. Jaganraj, "Identify crime detection using data mining techniques," *International Journal of Advanced Research in Computer Science and Software Engineering*, vol 3/8, pp. 977–983, Aug. 2013.

[3] M. Bakhshi, M. R. Feizi-Derakhshi, E. Zafarani, "Review and comparison between clustering algorithms with duplicate entities detection purpose," *International Journal of Computer Science Emerging Tech*, vol. 3, no. 3, pp. 108–114, 2012.

[4] M. Enache, M. Hulea and T. S. Letia, "A new approach in bloggers classification with hybrid of K nearest neighbour and artificial neural network algorithms by training neural network for construction of informatics offender profile," *Indian Journal of Science and Technology*, vol. 8, no. 3, pp. 237–246, Feb. 2015.

[5] Z. Eslami, M. Noroozi, S. K. Rad, "Provably secure group key exchange protocol in the presence of dishonest insiders," *International Journal of Network Security*, vol. 18, no. 1, pp. 33–42, Jan. 2016.

[6] F. S. Gharehchopogh, S. R. Khaze and I. Maleki, "A new approach in bloggers classification with hybrid of K nearest neighbour and artificial neural networks algorithms," *Indian Journal of Science and Technology*, vol. 8, no. 3, pp. 237-246, Feb. 2015.

[7] J. Herhalt, "Cyber crime - A growing challenge for governments," *KPMG International Issues Monitor*, vol. 8, pp. 1–21, July 2011.

[8] X. Li, X. Liang, R. Lu, X. Shen, X. Lin, H. Zhu, "Securing smart grid: Cyber attacks, countermeasures, and challenges," *IEEE Communications*, vol. 50, no. 8, pp. 38–45, Aug. 2012.

[9] S. Ni, M. Xie, Q. Qian, "Clustering based K-anonymity algorithm for privacy preservation, school of computer science and engineering," *International Journal of Network Security*, vol. 19, no. 6, pp. 1062-1071, Nov. 2017.

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

[11] C. Phua, K. Smith-Miles, V. Lee, R. Gayler, "Resilient identity crime detection," *IEEE Transactions on Knowledge and Data Engineering*, vol. 24, no. 3, pp. 533–547, 2012.

[12] S. Revathi, T. Nalini, "Performance comparison of various clustering algorithms," *International Journal of Advanced Research Computer Science and Software Engineering*, vol. 3, no. 2, pp. 67–72, Feb. 2013.

[13] G. Sehgal, K. Garg, "Comparisons of various clustering algorithms," *International Journal of Computer science and Information Techologies*, vol. 5, no. 3, pp. 3074–3076, 2014.

[14] T. Sanjana, C. M. Sheela and K. V. Narayana, "A survey on clustering techniques for big data mining," *Indian Journal of Science and Technology*, vol. 9, no. 3, pp. 1-12, Jan. 2016.

[15] J. J. Sheu, "Distinguishing medical web pages from pornographic ones: An efficient pornography websites filtering method," *International Journal of Network Security*, vol. 19, no. 5, pp. 839–850, Sept. 2017.

[16] P. Singh, A. Surya, "Performance analysis of clustering algorithms in data mining in Weka," *International Journal of Advances in Engineering & Technology*, vol. 7, no. 6, pp. 1866–1873, Jan. 2015.

[17] J. R. Sun, M. L. Shih, M. S. Hwang, "A survey of digital evidences forensic and cyber crime investigation procedure," *International Journal of Network Security*, vol. 17, no. 5, pp. 497–509, 2015.

[18] M. Sun, Y. Qin, J. Bao and X. Yu, "GPS spoofing detection based on decision fusion with a k-out-of-n rule, school of information science and engineering, south east university," *International Journal of Network Security*, vol. 19, no. 5, pp. 670–674, Sept. 2017.

[19] M. Uma, G. Padmavathi, "A survey on various cyber attacks and their classification," *International Journal of Network Security*, vol. 15, no. 5, pp. 390–396, Dec. 2011.

[20] C. Valentine, C. Hay, K. M. Beaver, T. G. Blomberg, "Through a computational lens : using dual computer criminology degree programs to advance the study of criminology and criminal justice practice," *Security Informatics*, DOI: 10.1186/2190-8532-2-2, Jan. 2013.

[21] Z. Zhan, M. Xu, S. Xu, "Characterizing honeypot-captured cyber attacks : Statistical Frame work and Case study," *IEEE Transactions on Information Forensics and Security*, vol. 8, no. 11, pp. 1775–1790, 2013.

[22] Z. Zhang, L. Yang, H. Li, and F. Xiang, "A quantitative and qualitative analysis-based security risk assessment for multimedia social networks," *International Journal of Network Security*, vol. 18, no. 1, pp. 43–51, Jan. 2016.

Biography

K. Veena received her M.E., (I.T) degree from Vinayaka Missions University, Salem, Tamilnadu in 2007 and B.E., degree from B.V.Bhoomraddi College of Engineering and Technology, Karnatak University, Karnataka.Her main interest is in Security issues regarding the safety of women.

K. Meena obtained her Ph.D., in Manonmaniam Sundaranar University, Tirunelveli. She is a full time Associate Professor at Vel Tech Dr. RR and Dr.SR Technical University. Her research interests are Pattern Recognition, Biometrics,Image Processing, Wireless Sensor Network, Cryptography and Network Security, High Speed Networks and Computer Networks.She is having 16 International Journal, 7 international Conferences and 11 National Conferences to her credit.

An Outlook on Cryptographic and Trust Methodologies for Clusters Based Security in Mobile Ad Hoc Networks

V. S. Janani and M. S. K. Manikandan
Department of Electronics & Communication Engineering, Thiagarajar College of Engineering
GST Road, Thiruparankundram, Madurai, Tamil Nadu 625015, India
jananivs@tce.edu

(Received Nov. 24, 2016; revised and accepted Mar. 31, 2017)

Abstract

Most of the researches in Mobile Ad Hoc Networks (MANETs) are closely related to security aspects and security issues. However, providing a security mechanism implicitly has been a major challenge in such ad hoc environment due to the dynamic movement of nodes. There are many security protocols as well as key management methods designed in a Public Key Infrastructure (PKI) to handle these MANET issues. To obtain a better understanding of various cryptographic and trust based security aspects, that forms an integral part of the solutions for issues in a clustered MANET, we provided a study on such security features. Through this study, security aspects such security attacks, security services, security challenges, and security solutions are described in a detailed manner.

Keywords: Clustering; Cryptography; MANET; PKI; Trust

1 Introduction

In traditional wireless networks, the existence of infrastructure enables secure communications between nodes on the network, over a limited geographical area [15]. Now-a-days, the demand for faster network setup without any access point or infrastructure is getting increased. Mobile Ad hoc Network (MANET) has been introduced as a solution for such requisites to provide communication over various applications [8]. MANET [14, 29] is defined an infrastructure less IP based cluster of mobile node or computing device, interrelated through multi-hop wireless links.

In MANET, the nodes possess a non-centralized administration system and so the nodes can join or leave freely, to obtain dynamic network topology. Every node in a MANET links to the nearby nodes over transmission range and may work as a router as well as host simultaneously. To communicate with a node, the source node forwards the data to the destination node through its neighboring nodes. Similar to the wired network, the neighbor node in MANET will perform as a router, which makes it a challenging task to detect malicious/legitimate nodes among the neighboring nodes.

Even though trust among the nodes is considered to perform co-operative communication within the network, MANET has more security threat while comparing with infrastructure-based wireless networks. Also, the dynamic environment, inadequate resources (*i.e.* battery power, bandwidth, storage, *etc.*), and lack of centralized monitoring make all communication layers in MANET vulnerable to various attacks. Therefore, MANETs must offer guaranteed for several security levels in order to have effective deployment and usage.

A MANET consists of mobile nodes with an autonomous system that may have gateway to an interface or function in isolation. The topology of the network may vary with respect to the continual node movement and the changes in transmission/reception parameters such as coverage patterns, power levels and interference levels of co-channel. Wherefore, MANETs have several salient characteristics [10, 16] as follows, which make MANETs more vulnerable than conventional networks.

- Infrastructure-less: The absence of static routers, centralized server, and other hardware infrastructures prevents the positioning of central host relationships. A distributed cooperative system is maintained in MANET to cope up with centralized functionalities.

- Wireless link usage: An adversary in wired network must passes through many defence lines at gateways and firewalls. Whereas, attacks on MANET can arise from various sources targeting any node in the network. Every node must be organized to secure against threats as a MANET does not have a clear defence line.

- Multi-hop: Hosts can act themselves as routers due

to the absence of centralized routers and gateways. Therefore, the packets follow multi-hop routes and move across distinct mobile nodes before receiving at their ultimate destination. Multi-hop feature presents a severe vulnerability because of the possible undependability of such mobile nodes.

- Node movement autonomy: Mobile nodes can be freely traversed in the network as they are usually autonomous units. This clearly shows that following down a specified mobile node in large-scale MANET cannot be completed easily.
- Amorphous: The nodes can join and leave the network unexpectedly due to dynamic node mobility and wireless connectivity. This leads topology changes with accidental link formation and breakage. This feature must take into account at any security solution.
- Memory and power limitation: The hosts in MANET are lightweight and have inadequate storage. These shortcomings make the network liable to energy starvation attack or sleep deprivation torture attack, where the attackers may aim some batteries of nodes to detach them. While designing the solutions towards security for MANETs, these features are also considered as a challenging constraint.

In this paper, we seek to provide a review on various methodologies for providing security in a cluster based MANET. We consider cryptography and trust as the two major dynamics that help in security establishment especially for cluster based ad hoc nodes.

2 Security Aspects in MANET

This section designates security aspects in MANET, which includes security attacks, security services and security challenges [9, 10, 17, 19, 20, 27].

2.1 Security Attacks on MANET

The MANET consists of miscellaneous nodes which may includes a malicious/ attacker node that affects the functionality of different MANET layers.

The Table 1 lists out the attacks at various MANET layers. The attacks mainly fall under two main classes: active attack or passive attack. The characteristics of MANET are vulnerable to the below mentioned attacks [5].

Active attacks: This type of attack tries to modify the protocol behavior by performing the operations like replication, modification, and deletion of interchanged data. It destroys or prevents message flow between the nodes. This can be collectively termed as Denial-of-Service (DOS) attacks, which completely block and damage the communication among the nodes.

Table 1: MANET attacks

MANET layers	Type of Attacks
Physical layer	Eavesdropping Jamming Active Interference
Link layer	Selfish node misbehavior DOS Attack Resource Exhaustion
Network layer	Black Hole Attack
	Wormhole Attack Routing Table Poisoning Attack Sleep Deprivation Attack Impersonation Attack Node Isolation Attack Location Disclosure Attack Rushing Attack Blackmail Attack The Invisible Node Attack (INA)
Transport layer	Session Hijacking
Application layer	Malicious code attacks
Multilayer attacks	Denial of Service Impersonation attacks Man-in-the-middle attacks

Passive attacks: This attack includes unauthorized snooping of information, packet eavesdropping and sometimes disabling a prime node from communication. This brings down the network and it contains: hidden channels, traffic analysis and unstable compromised keys.

2.2 Security Services on MANET

The most important security services that safeguard MANET resources from attacks are described as follows [30]:

- Authentication: It guarantees that a node is the one that has to be. Using this mechanism, only authorized nodes can communicate or transmit the data.
- Availability: This security service is employed to preserve the network resources obtainable to legitimate users. It also assures a reliable and appropriate use of data or the network.
- Data Confidentiality: The goal of this security service is to kept information confidential from disclosure [4] and it must be obtainable only to the intended party. This can be implemented via many data encryption methods.
- Data Integrity: The integrity of the data guarantees transmitted or communicated data is not being changed by any other mischievous node.

- Non-repudiation: In non-repudiation service, both sender and receiver would not be able to repudiate a transmitted message.
- Resilience to attacks: Even though a node is compromised partially, this security service makes it tolerant the functionalities of the network.

2.3 Security Challenges on MANET

Many complicated security challenges [11,13,21,23,28,33] that occurred in a MANET are addressed as follows:

- Dynamic Topology: In a MANET, establishing trust between the nodes is very complex as node may join or leave dynamically and changes frequently.
- Lack of Central Authority: Implementing security without any infrastructure or central authority in the network is a challenging job in a MANET.
- Insecure Environment: In a MANET, malicious node can attack and bargain the data while the nodes are moving randomly.
- Routing: In a MANET, routing protocols are most significant, where the nodes mobility varies very often. These protocols are employed for identifying the optimal path from source to destination node. Also, they are developed to exchange the information about routing [21].
- Multicasting: Traditional protocols of wireless networks do not suit one-to-many communication process named multicasting due to MANET characteristics [28,31]. An efficient protocol is required to meet various multicasting challenges.
- Energy Constraints: Mobile nodes will run with battery power, as to manage and avoid node termination. Energy management plays a vital role in MANET due to divers MANET challenges.
- Quality of Service (QoS): The major objective of the QoS is to offer better network services by accurately utilizing the resources of a network. Depending upon the user and application, QoS gathers bandwidth, delay, loss, etc to satisfy their tasks.
- Security: MANET is extremely susceptible to several security attacks, because of its existing key characteristics. It is very hard to accomplish security goals, where the intruders can easily damage the network. While manipulating security solutions, the distinctive features of MANET must be considered with higher priority.
- Clustering: In a MANET, the nodes are separated into virtual groups called clusters, to accomplish scalability even in the existence of high mobility in the network.

3 Security Mechanisms in MANETs

3.1 Cryptography

Generally, cryptography is considered as a powerful tool [6] introduced to construct and analyse various security protocols, by providing all the required network services. However, it can also be defined as a process by which plain text (original data) can be converted into cipher text (scrambled data) and vice versa, using secret keys. It can be classified into two types depending on the secret key used: *Symmetric/private key cryptography* (where the message is encrypted and decrypted with a single key) and *Asymmetric/public key infrastructure (PKI)* (where the information is processed by two different keys). Nevertheless, all these cryptographic techniques are the primitives of security, which can be widely utilized in both wired and wireless networks to provide confidentiality, authentication, integrity, and non-repudiation [4].

Most of the researches in MANET rely on the fact that there exist cryptographic mechanisms to secure keys, for various applications. Many researchers have suggested asymmetric cryptographic techniques to handle ad hoc protocols. But the infrastructure-less MANET makes it a challenging task especially during asymmetric signature verification. To get the better of the challenge, symmetric key techniques were proposed.

3.2 Public Key Management

To deploy PKI system in MANETs two main alternatives have been suggested as: distributed or non-centralized Certification Management, and self-organized PKI management. In distributed certificate management, the certification processes are supported by distributed Certificate Authorities (CA) that issue and validate certificates for each node.

Self- Organized key management have become a principal solution for any secure communication that incorporates the procedures and techniques to support the cryptographic keying relationships among certified parties. Besides, it establishes many services such as key initialization, key generation, key distribution, and key updating of the network. In key management, a key can be established [26] either using key agreement or key distribution protocols.

The key agreement protocols are characterized by the absence of trusted authorities responsible for key management, in which a key is constructed by two or more node collaboration. While, in a key distribution protocol a single node generates and distributes keys to other nodes in the presence of trusted authorities. The distribution protocol can be categorized into symmetric or asymmetric (certificate based, identity based) schemes, to make it suitable for ad hoc nature. Although key agreement schemes have not designed certainly for MANET, it fits the wireless environment. There are numerous key

Table 2: Key management methods in MANET

Schemes	Types
Symmetric Key Management	Distributed KeyPre Distribution Scheme Peer Intermediaries for Key Establishment Key Infection
Asymmetric Key Management	Self-Organized Key Management Secure and Efficient Key Management Private ID Based Scheme
Group Key Management (GKM)	Centralized Distributed Decentralized Simple and Efficient GKM Private Group Signature Key
Hybrid Key Management	Cluster Based Composite Scheme Zone-based scheme

management methods employed to accomplish greater security using cryptographic keys. Some of those key management methods in MANET are mentioned below in Table 2.

Moreover, the certificate based key distribution requires digital certificates signed by a trusted CA to bind public keys to authenticated nodes. These certificates encompass key materials, owner nodes identity and valid digital signatures, to make trust on the signer. In contrast with certificate based PKI scheme, identity based scheme uses nodes identity signed by a trusted entity as public key.

Most of the solutions introduced to cryptography were intended to secure data forwarding and routing mechanisms [6]. Moreover, due to the lack of any central administration in MANETs, key management has been a challenging issue. Certainly, this infrastructural role should be distributed among all nodes to form a key based infrastructure. Hence, the key management scheme of MANETs does not trust or rely on any stable CA, but indeed it should be self-organized and distributed.

4 Trust Models

Trust is one of important security characteristic that enables nodes to cope with the uncertain nature of MANET and consequently, trust calculation as well as management is difficult in MANET [2,7,24,34]. An untrustworthy node certainly has adverse affects the performance of the network. Therefore, calculating the trust level of each node has a promising influence on the security with which a node can be a part of a secure communication.

4.1 Trust Calculation Model

The trust calculation can be broadly classified into two types: Centralized and Decentralized trust models.

1) Centralized trust model:
Most of the centralized trust models assume one or more Trusted Third Party (TTP) as a central entity to compute and manage trust [7]. It is trusted by all nodes and is frequently employed for providing key management services. The TTP either calculates the trust for entire MANET or provides the initial trust value to each node. The centralized trust can be calculated by different methods:

Cluster based trust model: Trust is calculated by combining the initial trust obtained from the header node with the individual trust. This individual trust value may be based on the successful/unsuccessful experiences with the neighboring nodes during data communication.

Representative based trust model: Reputed representatives/agents are deployed by each node to assist the trust calculation in this model. To compute trust of neighboring nodes, each node verifies about those neighbors with their representatives. Final trust is calculated with the obtained trust value from the agent with the individual value.

Leader based trust model: A distributed trust is maintained at each group, where the group leader calculates the final trust based on the direct observations and the collective trust obtained from the group members.

2) Decentralized trust model:
Due to the lack of maintaining a global trusted entity in MANET, each node computes trust on its neighbors by itself, in decentralized model. Here the trust can be calculated by using any of the following three methods.

Direct trust: Each node observes the communication of its neighboring nodes and keeps a record of those communications within it. To compute trust, the trustor node weighs its own record with the record received from the trustee and other neighboring nodes. Direct trust can be computed by different ways as: packet routing and past-present observation methods.

Indirect trust: Decentralized trust can be calculated indirectly based on the recommendation of the neighboring nodes on a target node. This can be achieved either by voting method or by flooding the recommendation throughout the MANET.

Hybrid trust: This model takes advantage of the optimistic features of both direct and indirect trust models. It integrates direct observations and recommendations to compute trust effectively.

The absence of stable trust entities, resource limitations, frequent link failures and other security vulnerabilities makes the decentralized trust models a challenging one. To manage these issues, most of the methods proposed so far assumed to have a centralized trust entity.

4.2 Trust Application in MANET

From decades, cryptography has been considered as the most prominent methodology to secure the network from adversaries. It comprises of only an initial security check in terms of authentication, confidentiality, integrity and non-repudiation. Those methodologies, in fact provided only a partial solution from which an attacker node can easily impersonate. The threats that alter the credential security (soft security threats) cannot be eliminated completely with these methods. Trust has been widely applied in MANET not as a replacing methodology [12], but as an accessory to work against the opponents. Trust mechanisms and cryptography can be deployed together to provide a complete solution to the security threats in MANET [18].

5 Clustering Methodology

Clustering can generally be defined as the grouping of nodes in a network into an interrelated sub-structure [3]. In a MANET, a clustering scheme partitions the mobile nodes into virtual groups called clusters [1, 3, 32]. There are three main components in a cluster-based network: Cluster Head (CH), Cluster Members (CM) and Cluster Gateway [25]. Figure 1 shows typical cluster architecture in a mobile ad hoc network. The CH assists as a leader for its group, carrying out different cluster activities as packet forwarding, inter-intra communications clustering and so on. The CMs are ordinary nodes which reside in various clusters. A cluster gateway is nothing but an intermediate non- CH node that connects two adjacent clusters. The survey on clustering schemes evidently shows its achievement in MANET performance, especially in maintaining the topology. Some of the benefits of clustering in MANET are:

- Maximize the capacity of network by reusing existing resources. Similar set of frequency is employed only when two clusters are not adjacent and overlapped.
- Among adjacent clusters, CH and border nodes generate a virtual backbone for a beneficial routing.
- Minimize the storing information overhead by updating only the information of mobile nodes that relocated to another cluster.
- Decrease of control packet.
- Stability, simplification, and localization.

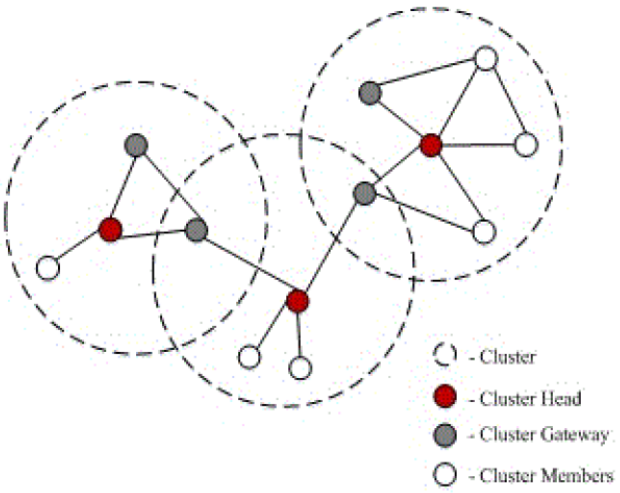


Figure 1: Cluster architecture in a MANET

5.1 Clustering Approaches in MANET

Clustering in MANET is performed based on different criteria as given below:

Minimized Dominating Set based clustering: This clustering approach is used to discover a minimum/weakly connected dominating set for a given network. It decreases the number of nodes that contributes in route search or maintenance of routing table and constructs CHs to proceed inter- cluster communication rapidly. *Example:* Connected Dominating Set (CDS) and weak CDS based clustering methods.

Low cost maintenance clustering: In order to minimize the clustering-based maintenance cost, a cluster infrastructure is provided for upper layer applications. *Example:* Least Cluster Change (LCC), Passive Clustering (PC), and 3-hop Between Adjacent Cluster head (3hBAC).

Mobility-aware clustering: Here, the mobility characteristic of the MANET nodes is considered for cluster construction and maintenance. This approach assigns the mobile nodes with low relative speed within a cluster to maintain the connection. *Example:* Mobility Based Metric for Clustering (MOBIC) and Distributed Dynamic Clustering Algorithm (DDCA).

Energy-efficient clustering: In order to proliferate the network lifetime, this approach either avoids or balance unnecessary energy consumption of mobile nodes. *Example:* Energy based Dominating Set and Identity based Load Balancing Clustering (IDLBC).

Power- aware clustering: To save the battery power in MANET, power aware clustering can be done by load-balancing, reducing the size of dominating set or by minimizing the consumption of transmission

energy. *Example:* Degree-Load-Balancing Clustering (DLBC).

Other-metrics-based clustering: Clustering can also be performed based on various metrics such as identity of nodes, size of cluster, degree of node, weight of cluster *etc.* *Example:* Weighted Clustering Algorithm (WCA), and On-Demand WCA.

5.2 Clustering Schemes from Security Perspective

A clustering scheme can be secured with various mechanisms as (1) cryptographic-based clustering (2) trust-based clustering and (3) hybrid clustering methods.

1) Cryptographic-based Clustering Methods:
The security of clustering operation against attackers has been increased with traditional cryptographic-based clustering methods. But, the insider attackers and compromised nodes remain undetected. This can be protected by using trust and reputation management methods. In MANETs, these methods have high overheads and inadequate resources. Therefore, secure clustering methods predominantly focus on defending the current CHs and choosing valid and accurate node as novel CH. Moreover, several security attacks can be accompanied against clustering. Following is the classification of attacks on clustering schemes [22] as

   - Clustering operation attacks;
   - Cluster maintenance; operation attacks;
   - Cluster component attacks.

   Cryptographic-based clustering methods employ cryptography for protecting networks against security threats. This offers security services like data privacy, digital signatures, and authentication. Depending on the key management techniques, the cryptographic security solutions are set to be high.

2) Trust-based Clustering Methods:
Trust-based clustering methods incorporate the trust management methods along with the clustering techniques. This can decrease the reputation management overheads. For each node, these methods accomplish the trust-based information. It further avoids the election of misbehaving nodes as cluster components. There are mainly two kinds of trust-based clustering method: pure and hybrid.

   Pure trust based clustering: This method comprises two main purposes: (1) enhancing the security of network by selecting reliable nodes as CHs, (2) minimize the trust management system overheads. This method is liable to numerous attacks like self-promoting attacks and bad mouthing. These security systems do not for entire protection against attackers and are susceptible to mischievous nodes and internal malicious nodes.

   Hybrid trust based clustering: These methods are the most difficult security solutions, which incorporate the cryptography-based techniques and reputation management schemes with clustering methods. This can protect against both internal and external attackers as it creates complex and strong solutions towards security. It also has the highest level of resource consumption.

6 Conclusion

This paper provides a brief introduction to the MANET and its key characteristics. In the subsequent section, the security aspects such as attacks, security services and security challenges are described in a detailed manner. The core domain of this research which comprises the security mechanisms, trust model and clustering approaches are explained with their appropriate examples in the following sections. The security mechanisms cover an overview of two predominant tools of wireless networks; cryptography and public key management.

References

[1] M. Alinci, E. Spaho, A. Lala, V. Kolici, "Clustering algorithms in MANETs: A review," in *IEEE Ninth International Conference on Complex, Intelligent, and Software Intensive Systems (CISIS'15)*, DOI: 10.1109/CISIS.2015.47, 2015.

[2] A. Ankit, A. K. Verma, "A review & impact of trust schemes in MANET," in *Proceedings of the International Conference on Advances in Information Communication Technology & Computing (AICTC '16)*, 2016.

[3] A. Bentaleb, A. Boubetra, S. Harous, "Survey of clustering schemes in mobile ad hoc networks," *Communications and Network*, vol. 5, no. 2, pp. 32-48, 2013.

[4] B. S. Bhawani, V. Pallapa, "Performance analysis of location privacy preserving scheme for MANETs," *International Journal of Network Security*, vol. 18, no. 4, pp. 736-749, 2016.

[5] W. Bing, J. Chen, J. Wu, M. Cardei, "A survey of attacks and countermeasures in mobile ad hoc networks," in *Wireless Network Security, Signals and Communication Technology*, pp. 103-135, 2007.

[6] W. Bing, J. Wu, M. Cardei, "A survey of key management in mobile ad hoc networks," in *Handbook of Research on Wireless Security*, vol. 8, no.3, pp. 48-66, 2007.

[7] J. H. Cho, C. I. Ray, K. S. Chan, "Trust threshold based public key management in mobile ad hoc networks," *Ad Hoc Networks*, vol. 44, pp. 58-75, 2016.

[8] M. Conti, C. Boldrini, S. S. Kanhere, E. Mingozzi, E. Pagani, P. M. Ruiz, M. Younis, "From MANET to people-centric networking: Milestones and open research challenges," *Computer Communications*, pp. 1-21, 2015.

[9] P. Di, S. Guarino, N. V. Verde, J. D. Ferrer, "Security in wireless ad-hoc networks: A survey," *Computer Communications*, vol. 51, pp. 1-20, 2014.

[10] D. Djenouri , L. Khelladi, N. BadXache, "A survey of security issues in mobile ad hoc networks," *IEEE Communications Surveys*, vol. 7, no. 4, pp. 2-28, 2005.

[11] A. Dorri, R. K. Seyed, K. Esmaeil, "Security challenges in mobile ad hoc networks: A survey," *International Journal of Computer Science & Engineering Survey (IJCSES'15)*, vol.6, no.1, pp. 15-29, 2015.

[12] P. Gera, G. Kumkum, G. M. Manoj, "Trust-based multi-path routing for enhancing data security in MANETs," *International Journal of Network Security*, vol. 16, no. 2, pp. 102-111, 2014.

[13] P. Godwin, D. R. Srinivasan, "A survey on MANET security challenges, attacks and its countermeasures," *International Journal of Emerging Trends & Technology in Computer Science*, vol. 3, no. 1, pp. 274-279, 2014.

[14] C. J. Hee, A. Swami, R. Chen, "A survey on trust management for mobile ad hoc networks," *IEEE Communications Surveys & Tutorials*, vol. 13, no. 4, pp. 562-583, 2011.

[15] A. Hinds, M. Ngulube, S. Zhu, A. A. Hussain, "A review of routing protocols for mobile Ad-Hoc networks," *International Journal of Information and Education Technology*, vol. 3, no. 1, pp. 1-5, 2013.

[16] C. Jianmin, J. Wu, "A survey on cryptography applied to secure mobile ad hoc networks and wireless sensor networks," *Handbook of Research on Developments and Trends in Wireless Sensor Networks: From Principle to Practice*, pp. 262-289, 2010.

[17] A. Kartit, K. I. Hamza, B. Mohamed, "Improved methods and principles for designing and analyzing security protocols," *International Journal of Network Security*, vol.18, no.3, pp. 523-528, 2016.

[18] A. Kumar, G. Krishna , A. Alok, "Design and analysis of lightweight trust mechanism for secret data using lightweight cryptographic primitives in MANETs," *International Journal of Network Security*, vol. 18, no. 1, pp. 1-18, 2016.

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

[20] C. T. Li, M. S. Hwang, C. Y. Liu, "An electronic voting protocol with deniable authentication for mobile Ad hoc networks," *Computer Communications*, vol. 31, no. 10, pp. 2534–2540, June 2008.

[21] C. Mahima, M. Ahmad, M. Waseem, "Review on MANET: Characteristics, challenges, imperatives and routing protocols," *International Journal of Computer Science and Mobile Computing*, vol. 3, no. 2, pp. 432-437, 2014.

[22] M. Maleknasab, M. Bidaki, A. Harounabadi, "Trust-based clustering in mobile ad hoc networks: Challenges and issues," *International Journal of Security and Its Applications*, vol. 7, no. 5, pp. 321-342, 2013.

[23] K. Mohit, R. Mishra, "An overview of MANET: History, challenges and applications," *Indian Journal of Computer Science and Engineering*, vol. 3, no. 1, pp. 121-125, 2012.

[24] Z. Movahedi, H. Zahra, B. Fahimeh, P. Guy, "Trust-distortion resistant trust management frameworks on mobile ad hoc networks: A survey," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 2, pp. 1287-1309, 2016.

[25] M. U. Muhammad, M. Amjad, S. Houbing, "SeCRoP: Secure cluster head centered multi-hop routing protocol for mobile ad hoc networks," *Security and Communication Networks-Wiley*, vol. 9, no. 16, pp. 3378-3387, 2016.

[26] K. Pratibha, N. Goyal, "Survey of various keys management techniques in MANET," *International Journal of Emerging Research in Management &Technology*, vol. 4, no. 6, pp. 176-178, 2015.

[27] K. Praveen, P. C. Sekhar, N. Papanna, N. B. B. Bhushan, "A survey on MANET security challenges and routing protocols," *International Journal of Computer Technology and Applications*, vol. 4, no. 2, pp. 248-256, 2013.

[28] G. Priyanka, V. Parmar, R. Rishi, "Manet: Vulnerabilities, challenges, attacks, application," *International Journal of Computational Engineering & Management (IJCEM'11)*, vol. 11, pp. 32-37, 2011.

[29] A. Rana, D. Sharma, "Mobile ad-hoc clustering using inclusive particle swarm optimization algorithm," *International Journal of Electronics and Information Engineering*, vol. 8, no. 1, pp. 1-8, 2018.

[30] S. Sen, J. A. Clark, J. E. Tapiador, "Security threats in mobile Ad hoc networks," in *Security of Self-Organizing Networks: MANET, WSN, WMN, VANET, Al-Sakib Khan Pathan, Ed. Boca Raton, Florida: Taylor & Francis*, 2010.

[31] M. Stanek, "A note on security protocol for multicast communications," *International Journal of Network Security*, vol. 14, no. 1, pp. 59-60, 2012.

[32] S. Victor, R. Ayman, H. Marques, J. Rodriguez, S. Vahid, R. Tafazolli, "A survey on clustering techniques for cooperative wireless networks," *Ad Hoc Networks-Springer*, vol. 47, pp. 53-81, 2016.

[33] V. K. Vishakha, N. K. Bhil, "Mobile ad-hoc network (MANET) and its security aspects," *International Journal of Advance Research in Computer Science and Management Studies*, vol. 3, no. 8, pp. 145-149, 2015.

[34] G. Xu, Y. Zheng, "A survey on trust evaluation in mobile ad hoc networks," in *Proceedings of the 9th EAI International Conference on Mobile Multimedia Communications*, pp. 140-148, 2016.

Biography

V. S. Janani received her B.E. degree in Electronics and Communications Engineering from Anna University in 2009 and M.E degree in Embedded System Technologies from Anna University in 2011.Since 2012 she has been pursuing her PhD degree in the department of Electronics and Communication Engineering at Thiagarajar College of Engineering.

M. S. K. Manikandan received his B.E. degree in Electronics and Communications Engineering from National Institute of Technology. Trichy, Bharathidasan University in 1998 and M.E degree in Communication System from Madurai Kamaraj University in 2000. He received his PhD in Wireless Communication from Anna University; Chennai in 2010.He has been working as associate professor in Thiagarajar college of Engineering for years. He has been serving as reviewer for several IEEE conferences, Springer, IET and other international journals.

A Credible Mechanism of Service about Data Resource in Cloud Computing

Yunfa Li, Yangyang Shen and Mingyi Li
(Corresponding author: Yunfa Li)

Key Laboratory of Complex Systems Modeling and Simulation, School of Computer, Hangzhou Dianzi University
Xiasha Higher Education Zone, Hangzhou, Zhejiang Province,310018, China
(Email: yunfali@hdu.edu.cn)

(Received Feb. 25, 2017; revised and accepted May 21, 2017)

Abstract

In order to protect the security of service in cloud computing, we propose a credible mechanism of service for data resource. In the mechanism, we first build a system model and put forward its implementation process. Then, we design an encryption method used in the system model. On this basis, we discuss the function of the third party trusted platform. All of these constitute a credible mechanism of service about data resource in cloud computing. In order to verify the feasibility of the mechanism, a series of experiments have been done. The experimental results show that it is feasible to ensure the security of service about data resource in cloud computing.

Keywords: Credible; Cloud Computing; Data Resource; Service; System Model

1 Introduction

With the rapid development of virtualization technology, cloud computing begin to be widely used in data processing, data analysis. Data are usually stored to cloud server by more and more users, which leads to the scale of data resource is becoming larger and larger. In this situation, data resource are faced with more and more network attacks and the security protection of data resource is also becoming more and more difficult. In general, some traditional security maintenance methods are mainly focused on the access control strategy of data resource, the encryption and decompression method of data file, the identity authentication mechanism of data resource, data signature and so on. These technologies can play an important role in the maintenance of data security, however, as for the virtualized computing environment, these techniques and these methods exist obvious disadvantage in the maintenance of security about data resource. The main reason is that these techniques and these methods cannot make sure if the service is provided by data resource or not, which will result in the leakage or tampering of data resource information. In this state, it is very important to research the credible mechanism of data service in cloud computing [12].

Based on the above description, we will study the service mode of data resource in cloud computing in this paper. The purpose is to explore a credible mechanism of the service about data resource and maintain the security of data resource in cloud computing. Therefore, this paper is organized as follow: Section 2 overviews the literature on the security of service in cloud computing. Section 3 will proposes a credible mechanism based on the service of data resource in cloud computing. Section 4 describes a series of experiments and analyzed the results. Section 5 is the conclusion of this paper.

2 Related Works

With the rapid development of the technology of cloud computing, the scale of data resource is becoming bigger and bigger, and theirs application is becoming wider and wider. In the situation, theirs security protection is becoming more and more difficult. In order to resolve these questions, some people carried out a series of research and achieved some results. These results can be described as follows.

In [2], the authors first analyzed the challenge about the intrusion severity analysis problem for the overall security of clouds. Then, the authors presented a novel method to address this challenge and analyzed the degree of intrusion detection in clouds. On this basis, a rigorous evaluation method was presented to assess the effectiveness and feasibility of their proposed method for clouds.

The security problems faced by cloud computing and the existing security technology are described in [15]. In the paper, the authors first discussed a series of challenges faced by the security of cloud computing. Then, the authors indicated there are many methods to improve the security of cloud. At last, the authors stressed there is no one precise solutions to resolve the security of cloud

and there is facing many challenges and difficulties in the future.

In [11], the authors studied three auditing schemes for stored data including the public auditing scheme with user revocation, the proxy provable data possession and the identity-based remote data possession checking. All three mechanisms claimed that their schemes satisfied the security property of correctness. It is regretful that this comment shows that an active adversary can arbitrary alter the cloud data to generate the valid auditing response which can pass the verification. Then, they discussed the origin of the security flaw and proposed methods to remedy the weakness.

Various security challenges, vulnerabilities, attacks and threats that hamper the adoption of cloud computing are described in [17]. In the paper, the authors first provided a state-of-art survey on cloud security issues and challenges. Then, the authors explored various cloud services and what they provide as well as analyzed the security concern on each provider. At last, the authors stressed existing schemes that counter the security issues in an efficient, and cost saving manner and proposed the 3-tier security architecture for better security enhancement of cloud security.

In [7], the authors first discussed institutions and institutional evolution in the cloud industry. Then, they examined the forces and nature of institutional changes in the cloud industry. In the following section, a section on discussion and implications are described. At last, some comments are concluded.

In [8], the authors shown that Seo *et al.*'s certificateless encryption is insecure because it binds only a user's partial public key with the user's identity. An adversary can replace the remaining public key of the user to launch man-in-middle attacks. they also presented an improvement using the technique developed by Schnorr.

The security of data sharing in cloud computing is described in [6]. In the paper, an effective, scalable and flexible privacy-preserving data sharing scheme is proposed in the cloud. In order to preserve privacy and guarantee data confidentiality, a cryptographic primitive, named ciphertext policy attribute-based encryption (CP-ABE) is employed and an identity-based encryption (IBE) technique is considered.

In [9], the authors first described how to constitute an integrated security cloud platform by using the fusion technology. Then, they explained some security threats faced the cloud infrastructure. According to user's demand, the solutions to these problems are presented in detail. Finally, the technology used in the secure cloud platform is confirmed.

In [5], the authors described an efficient three-party password authenticated key exchange (3PAKE) protocols based on LHL-3PAKE proposed by Lee *et al.* The 3PAKE requires neither the server public keys nor symmetric cryptosystems such as DES. The formal proof of security of their 3PAKE is based on the computational Diffie-Hellman assumption in the random oracle model along with a parallel version of the proposed 3PAKE.

Privacy cheating discourages and secure audit protocols are proposed in [20]. Its purpose is to resolve the security and privacy problem faced by data store and computing. In the protocol, the security audit is implemented by using the technology of authentication, verification and probability sampling, which is used to verify the data stored in the data storage.

In [18], the authors presented a solution to monitor and enforce the fulfillment of secSLAs. In the solution method, a comprehensive secSLA model and a secSLA management methodology are proposed, firstly. Then, an automated remediation process for potential and actual secSLA violations is introduced. In [13], the authors first described the security challenges of data in cloud computing. Then, they presented some solutions for these challenges to overcome the risk involved in cloud computing. At last, some advanced encryption techniques and some proper key management techniques are proposed in this paper for the security of data. In [10], the authors first discussed the challenges and described the solutions for protecting big data in cloud computing. Then, they proposed an architecture named $MetaCloudDataStorage$ to protect the security of big data in cloud computing.

In order to resolve the privacy requirements of large-scale WSNs and promote energy-efficient collection of big sensor data, the authors proposed a scalable privacy-preserving big data aggregation method in [21]. In this method, sensor nodes are divided into clusters, firstly. Then, sensor data is modified by each node according to the privacy-preserving configuration message. At last, aggregated results are recovered by the sink to complete the privacy-preserving big data aggregation.

In order to research the security auditing of cloud computing, Hassan [14] first analyzed three perspectives which include user auditing requirements, the technical approaches of security auditing and the provider capabilities of current cloud service. On the basis, specific auditing issues are divided into two categories. At last, a number of techniques available to address user auditing concerns are described in the data auditing area.

In order to resolve the problem of privacy preservation, Yousra *et al.* [1] first discussed the procedures of data collection. Then, a novel anonymous technique is proposed. In the anonymous technology, a motivation model is presented and some corresponding flowcharts are analyzed. At last, the evaluation criteria of the technology are proposed.

In order to resolve the problem of data-integrity verification, Anirudha *et al.* [16] first optimize an existing third party auditing protocol. Then, a protocol to perform efficient block-level and fine-grained dynamic-data update operations is proposed on data stored in cloud computing. At last, the extensive security and performance analysis is shown.

In [4], the authors show that in the Zhu-Jiang scheme the group manager cannot complete his computational task in the registration phase. They stress that, from the

practical point of view, the mechanism that the group manager has to re-encrypt all data after a member is revoked, is not generally acceptable because it is very inefficient.

In order to study the security of data storage in cloud computing, Naresh *et al.* [19] first provided service models of cloud, deployment models and variety of security issues in data storage in cloud environment. Then, possible solutions for the data storage are described which can provide privacy and confidentiality in cloud environment.

In [3], the authors define a Storage Correctness and Fine-grained Access Provision ($SCFAP$) scheme, that provides the user an exclusive access through the use of a hierarchical structure which is a combination of users unique and common attributes. Moreover, they deploy the concept of Token Granting system that allows the users to verify the correctness of outsourced data without the retrieval of the respective files.

Although the above some methods and technologies can resolve some security questions in cloud computing, there is a lot of security problems needed to be resolved because the service mode of data resource are various and the service environment is virtual. Therefore, there are many disadvantages for the existing security technology and methods. In order to overcome these disadvantages, we present a credible mechanism of service about data resource in cloud computing.

3 The Credible Mechanism of Service

In this section, we will propose a credible mechanism in cloud computing, which is based on the service of data resource. In this mechanism, we first build a system model and put forward its implementation process. Then, we design an encryption method used in the system model. On this basis, we discuss the function of the third party trusted platform. All of these constitute a credible mechanism of service about data resource in cloud computing. The process can be described as follows:

3.1 System Model

In cloud computing, all services provided by data resource to corresponding users are through the cloud server. At first, the user sends some data manipulation commands and requests to the cloud server, who wants to get the service of data resource. Secondly, the cloud server provides data resource services to the user through the virtual machine monitor. The virtual machine monitor plays a key role in controlling and managing the resource scheduling and service process. However, the services of data resource are usually faced with many security threats under the control and management methods. Some hackers or illegal users always want to get all kinds of information about data resource and destroy the system through a variety of ways. In order to achieve certain ulterior purpose, the hackers (or illegal users) usually use the following methods: create a large number of illegal data files, falsify the data resource, steal data, and send a large number of waste data. All these methods that the hackers used are related to the operation about data resource. In the face of a series of security problems in cloud computing, we need to construct the third party trusted platform (TTP) in cloud computing, which is used to enhance the security of service of data resource. Therefore, the system model is shown in Figure 1.

In this system model, the user is only responsible for providing the operation application to the cloud server. And the cloud server can provide the necessary data resource service according to user's requirement, such as modify, add, delete, insert and search. In the model, we suppose the cloud server contains all the data resource which is required by user and the information transmission is followed SSL (Secure socket layer) or TLS (Transport later security) protocols among the cloud service, the users and the third party trusted platform. The implementation processes can be described as follows:

Step 1: The user first encrypts a data file F by using a symmetric key K_D. Then, the encrypted file $K_D(F)$ is sent to the cloud server and the third party trusted platform.

Step 2: The symmetric key K_D is encrypted by using a public key K_P, firstly. Then, the private key K_S is signed by user. At last, the encrypted symmetric key $K_P(K_D)$ and the signed private key K_S are sent to the cloud server and third-party trusted platform by SSL or TLS.

Step 3: The cloud server will verify its identity after it received the private key K_S, which is sent by the user. If the authentication between the user and the cloud server is successful, the cloud server will receive the encrypted file $K_D(F)$ and the encrypted symmetric key $K_P(K_D)$. After that, the cloud server decodes the encrypted symmetric key $K_P(K_D)$ by the private key K_S and get a data file F^*. If the authentication is not successful, it shows error, go to Step 11.

Step 4: The cloud server encrypts the data file F^* with a symmetric key K_M, and then send encrypted file $K_M(F^*)$ to the third party trusted platform.

Step 5: The cloud server will encrypt the symmetric key K_M by using a public key K_{pc}, and then send the encrypted file $K_{pc}(K_M)$ to third party trusted platform by SSL or TLS.

Step 6: The third party trusted platform will apply for the private key K_{ps} to the cloud server after he receives the $K_{pc}(K_M)$.

Step 7: After the cloud server receives the application from the third party trusted platform, it will sign the private key K_{ps} and sends it to the third party trusted platform by SSL or TLS.

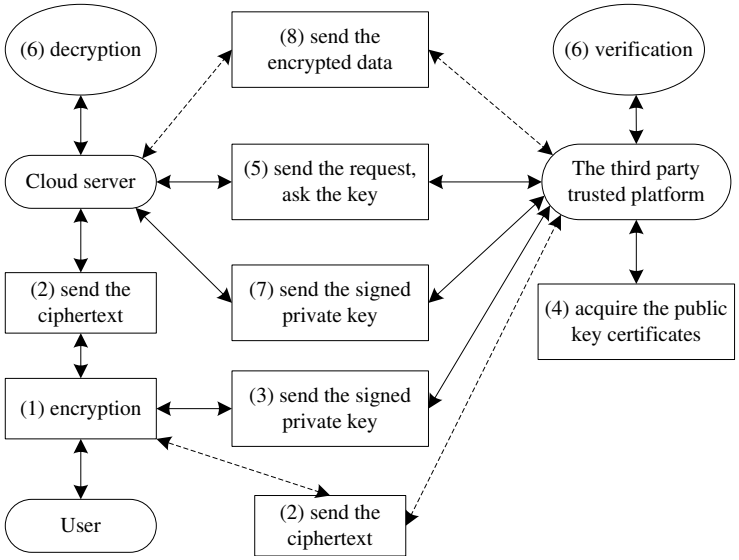


Figure 1: The system model of the credible mechanism of service about data resource

Step 8: The third party trusted platform will verify its identity after he receives the signature private key K_s. If the user authentication is successful, the third party trusted platform will decrypt the encrypted symmetric key $K_P(K_D)$ and get the symmetric key K_D. Then, the third party trusted platform will decrypt the encrypted file $K_D(F)$ and get a data file F by using the symmetric key K_D. If the authentication is not successful, it shows error and goes to Step 11.

Step 9: The third party trusted platform will verify the identity of the private key after it receives the signature private key K_{ps}, If the authentication about the cloud server is successful, the third party trusted platform will decrypt the received encryption file $K_{pc}(K_M)$ and get the symmetric key K_M. Then, it will decrypt the encrypted file $K_M(F^*)$ by using the symmetric key K_M and get the data file F^{**}, which is received from the cloud server. If the authentication is not successful, it shows error and goes to Step 11.

Step 10: The third-party trusted platform will verify whether F'' and F^{**} is uniform or not. If the two data files are same, it shows that the communication service between the users and the cloud server is credible. If the two data files are not same, it shows that the communication service between the users and the cloud server is not credible.

Step 11: End.

3.2 Encryption Method

In the credible mechanism of service for data resource, two kinds of different encryption methods are used. One is the symmetric encryption method and the other is the asymmetric encryption. In the symmetric encryption method, the *AES* Standard Advanced (Encryption) is used for encryption. The Standard is also named as the *Rijndael* encryption method, which belongs to a block encryption method. The encryption process is carried out in a 4×4 matrix of bytes and this matrix in cryptography theory is called "a state". The initial value is a plaintext block and the encryption cycle of the *AES* contains four operations: namely *AddRoundKey*, *SubBytes*, *ShiftRows* and *MixColumns*. The *AddRoundKey* operation is to make *XOR* operation for each byte in the matrix with the second round of the secret key (round key). And each sub key is generated by using the key generation scheme. The *SubBytes* operation is to replace each byte to the corresponding byte by using a lookup table, which is mainly through a nonlinear replacement function. The *ShiftRows* operation is to mix each column four bytes by using linear transformations. And the *MixColumns* operation will be omitted and replace by another *MixColumns* operation in the last encrypted loop.

In the asymmetric encryption algorithm, each entity (such as: user, cloud server *etc.*), generates a pair of keys $\langle U_k, V_k \rangle$, where U_k denotes the public key of entity U, V_k denotes the private key. In the asymmetric encryption algorithm, we will widen the *RSA* scheme to a scheme that employs the general linear group of order h with values selected randomly from the ring of integer $mod(n)$ where n is a product of two large prime numbers. The integer is coprime with n form a group under multiplication $mod(n)$ of order $g(n)$, inverse square matrices of rank h on the ring of integer $mod(n)$ will generate a group of order to this group, and this is unknown in the general scheme. But, in the general scheme where n is a product of two distinct prime numbers we can find the order of this group by the following theorem:

Theorem 1. *Assume that* $n = p \times q$ *is the product of two large prime numbers, and suppose that* g *is the general*

linear group of $h \times h$ matrices over Z_n. Then $g = (p^h - p^0)(p^h - p^1)\cdots(p^h - p^{h-1}) + (q^h - q^0)(q^h - q^1)\cdots(q^h - q^{h-1})$.

Proof. Each matrix $x \in g$ decreased to two matrices x_p and x_q such that x_p and x_q are $h \times h$ matrices on the members z_p and z_q such that $x_p = xmodp$, $x_q = xmodq$. Actually, a mapping: $f : g(n,h) \to g(p,h) \oplus g(q,h)$, is the ring identical of a two rings. □

In the asymmetric encryption algorithm, The whole processes include three phases, namely the key generation phase, the encryption phase and the decryption phase. Each phase is described as follows, respectively.

1) The key generation phase:

   Step 1: Entity U randomly and secretly chooses two large primes p and q, and compute $n = p \times q$.

   Step 2: Entity U computes $\psi(n) = (p-1)(q-1)$.

   Step 3: Entity U computes $g(n,h) = (p^h - p^0)(p^h - p^1)\cdots(p^h - p^{h-1}) + (q^h - q^0)(q^h - q^1)\cdots(q^h - q^{h-1})$.

   Step 4: entity U selects a random integer r, such as $1 < r < n$ and $gcd(r, \psi(n)) = 1$ and $gcd(r, g(n,h)) = 1$ (where r should be a small integer)

   Step 5: Entity U computes e such as $r \cdot e \equiv 1 mod \psi(n)$ and $1 < e < \psi(n)$.

   Step 6: Entity U computes d such as $d \cdot e \equiv 1 mod g(n,h)$ and $1 < e < g(n,h)$.

   Step 7: Entity U gets the public key $U_k \leftarrow (e,n)$

   Step 8: Entity U gets the private key $V_k \leftarrow (r,d,n)$

2) The encryption phase:
   Suppose entity M needs to send message m to entity U (represents m as an integer in the range of $0 < m < n$).

   Step 1: Entity U should send his public key to entity M.

   Step 2: Entity M will encrypt m as $c = ((m^e mod n)^e mod n)$.

   Step 3: Entity M will send c to entity U.

3) The decryption phase:

   Step 1: Entity U will decrypt the received message as: $m = ((c^r mod n)^d mod n)$.

3.3 The Third Party Trusted Platform

The main functions of the trusted third party platform include: (1) Receiving the private key of user. (2) Possessing the public key certificate which is authorized by the user. (3) Sending a private key to the cloud server. (4) Verifying the credibility of data that the cloud server receives or transmits. By using the third party trusted platform, we can get that if the communication service between the users and the cloud server is credible or not. Therefore, the section will describe the four aspect functions.

1) Receiving the private key of user:
   In order to hold the private key of each user in cloud computing, the third party trusted platform needs to constantly modify the database of the cloud user. All important information for each user are included in the database of the cloud user, such as the private key, the digital certificate of user, the key and the hash value of the cloud server.

2) Possessing the public key certificate which is authorized by the user:
   Once the third party trusted platform received the encrypted private key sent from the user, it possesses the public key certificate authorized by the user. In order to obtain the public key certificate, the user must use the private key to sign and then encrypt the signed information. On the basis of these steps, the user will send the signed and encrypted information to the third party trusted platform. For example, $E_{v_k \to user}\langle E_{U_k \to TTP}(K_s)\rangle$, where U_k denotes the public key of the user, v_k the private key of the user. Therefore, if the third party trusted platform wants to verify the user's identity, it needs to decrypt the signed and encrypted information, firstly. Then, it can verify the user's identity. The whole verification processes can be described as follows: $D_{U_k \to user}(E_{v_k \to user}\langle E_{U_k \to TTP}(K_s)\rangle) = E_{U_k \to TTP}(K_s)$.

3) Sending the private key to the cloud server:
   Before providing the service for the user, the trusted third party platform needs to verify the identity of the cloud server. The basic processes can be described as follows: $D_{U_k \to user}(E_{v_k \to user}(F_{K_D}) = H(M_i))$, where H is a cryptographic hash function. The main function of H is to encrypt the data blocks m_i of the data file F and generate a 160 bit hash value. Once the identity of the cloud server is authenticated, the third party trusted platform will take out the corresponding private key of the user from its own database and encrypt the private key by using the *RAS* encryption method. After that, the encryption data information is sent to the cloud server.

4) Verifying the creditability of the received or sent data:
   If the third party trusted platform receives the signature private key K_s, it will verify the identity of the user. If the authentication about the identity of the user is successful, it will decrypt the encrypted symmetric key $K_P(K_D)$ and get the symmetric key K_D. Then, it will decrypt the encrypted file $K_D(F)$ and get a data file F^{**} by using the symmetric key K_D. If it receives the signature private key K_{ps}, it will verify the identity of the cloud server. If the authentication about the identity of the cloud server is successful, it will decrypt the received encryption file $K_{pc}(K_M)$ and get the symmetric key K_M. Then, it will decrypt the encrypted file $K_M(F^*)$ by using the

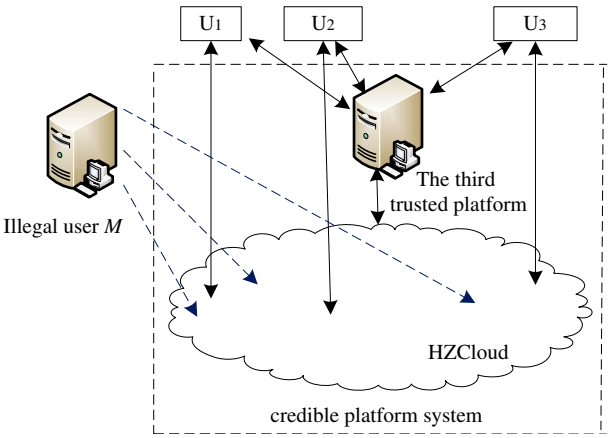


Figure 2: The overall framework of experiment

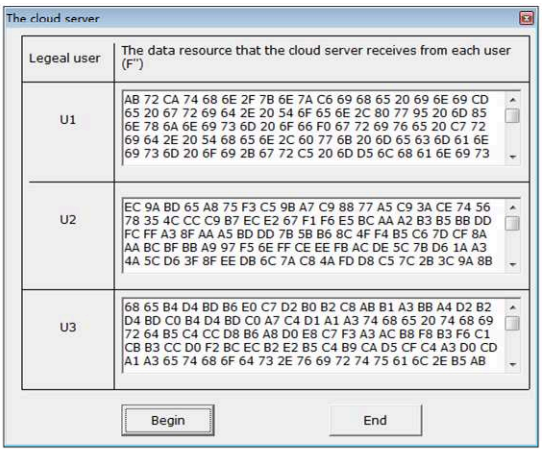


Figure 3: The data resource that the cloud server receives from each user

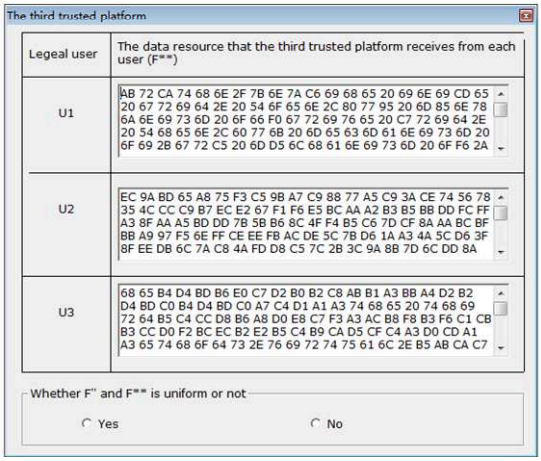


Figure 4: The data resource that the third trusted platform receives from each user

symmetric key K_M and get the data file F^{**}. At last, the third-party trusted platform will verify whether F'' and F^{**} is uniform or not. If the two data files are same, it shows that the communication service between the users and the cloud server is credible. Otherwise, it shows that the communication service between the users and the cloud server is incredible.

4 Experiment and Result Analysis

In order to verify the validity of this mechanism, we first carried out a series of experiments. On this basis, the experimental results and the advantages are analyzed. The whole processes are described as follows.

4.1 Experiment

In our experiments, we first constitute a credible platform system by using our proposed credible mechanism of service about data resource which is based on *HZCloud* cloud computing environment. In the credible platform system, it includes the cloud server, the third party trusted. Moreover, we built three legal users (namely, U_i(i=1, 2, 3)) and each legal user can use a different virtual machine which is respectively virtualized from the cloud server. Each legal user can submit some application to the cloud server and can get corresponding service of data resource. At the same time, there is an illegal user M who wants to steal the sent data resource. The overall framework of experiment is shown in Figure 2.

By using the credible platform system, we can get the state that the cloud server receives data resource from each user in the third trusted platform. The data resource that the cloud server receives from each user are shown in Figure 3, respectively.

Moreover, we can also get the state that the third trusted platform receives data resource from each user. The data resource that the third trusted platform receives from each user are shown in Figure 4, respectively.

In addition, we also get the data resource that the illegal user M steals. The result is shown as Figure 5.

4.2 Result Analysis

Comparing the "The data resource that the cloud server receives from each user" column of Figure 3 with the "The data resource that the third trusted platform receives from each user" column of Figure 4, we will find that the data resource that the cloud server receives from each user is the same as the data resource that the third trusted platform receives from each user, respectively. In addition, we can also find the data resource that the illegal user M receives from each user all are digital gibberish. These show that the cloud server can efficiently receives the data resource from each user and the illegal user M cannot decrypt the encrypted data resource although he

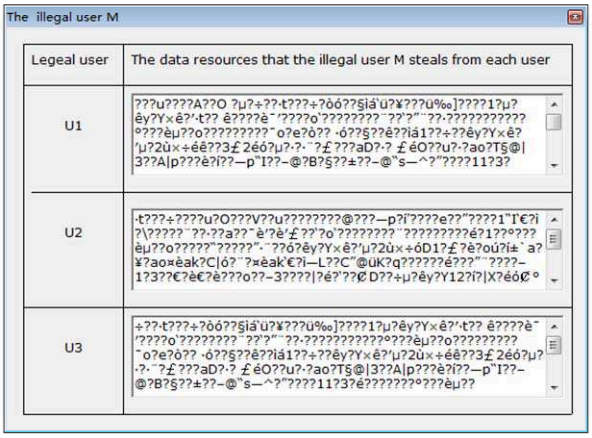


Figure 5: The data resource that the illegal user M steals from each user

can steals data resource from each user. The main reason that these situations generate is that our proposed encryption method is used.

4.3 Advantage Analysis

1) The key range of the described asymmetric encryption algorithm in our paper is considerable. It means that it can be large enough to use by matrices of high level of ranks. The key range for instance in the *RSA* scheme is $\psi(n) = (p-1)(q-1)$. But, in our described asymmetric encryption algorithm, the key range is of length $g(n,h)$. So, the differences between the two ranges are obvious.

2) The described asymmetric encryption algorithm in our paper can be used with Hill cipher method to obtain more intractable encryption system. Moreover, the described asymmetric encryption algorithm can be employed using a subgroup instead of a full value of $g(n,h)$, since the $gcd(r, g(n,h)) = 1$. Thus, the described asymmetric encryption algorithm in our paper will give more flexibility to entity to use more than one technique.

3) The intractability of the integer factoring of the modulus n in the described asymmetric encryption algorithm stays as same as in the *RSA* scheme.

From the above result analysis and the above advantage analysis, we can get that the data resource service between the users and the cloud server is credible.

5 Conclusion

In cloud computing, the access of user is becoming more and more frequently own to the scale growth of the data resource. The network attacks that users and cloud server confront are also becoming more and more frequently. Under the situation, the security protection of data resource in cloud computing becomes very difficult. The traditional network security methods can play a certain role in maintenance the security of data resource. However, these traditional network security methods are also facing many difficulties in the face of virtualized computing environment. In this situation, we propose a credible mechanism of service for data resource in cloud computing. In order to verify the feasibility of the mechanism, a series of experiments have been done. The experimental results show that it is feasible to ensure the security of service about data resource in cloud computing.

Acknowledgments

This study was supported by the National Natural Science Foundation of China (no. 61472112). The authors gratefully acknowledge the anonymous reviewers for their valuable comments.

References

[1] Y. A. A. S. Aldeen, M. Salleh, and Y. Aljeroudi, "An innovative privacy preserving technique for incremental datasets on cloud computing," *Journal of Biomedical Informatics*, vol. 62, pp. 107–116, 2016.

[2] J. Arshad, P. Townend, and J. Xu, "A novel intrusion severity analysis approach for clouds," *Future Generation Computer Systems*, vol. 29, pp. 416–428, 2013.

[3] B. Balusamy, P. V. Krishna, G. S. T. Arasi, and V. Chang, "A Secured Access Control Technique for Cloud Computing Environment Using Attribute Based Hierarchical Structure and Token Granting System," *International Journal of Network Security*, vol. 19, no. 4, pp. 559–572, 2017.

[4] Z. Cao, C. Mao, and L. Liu, "Analysis of One Secure Anti-Collusion Data Sharing Scheme for Dynamic Groups in the Cloud," *International Journal of Electronics and Information Engineering*, vol. 5, no. 2, pp. 68–72, 2016.

[5] T. Y. Chang, M. S. Hwang, and W. P. Yang, "A communication-efficient three-party password authenticated key exchange protocol," *Information Sciences* , vol. 181, pp. 217–226, 2011.

[6] X. Dong, J. Yu, Y. Luo, Y. Chen, G. Xue, and M. Li, "Achieving an effective, scalable and privacy-preserving data sharing service in cloud computing," *Computers and security*, vol. 42, pp. 151–164, 2014.

[7] N. Kshetri, "Privacy and security issues in cloud computing: The role of institutions and institutional evolution," *Telecommunications Policy*, vol. 37, pp. 372–386, 2013.

[8] L. Liu, W. Kong, Z. Cao, and J. Wang, "Analysis of one certificateless encryption for secure data sharing in public clouds," *International Journal of Electronics and Information Engineering*, Vol. 6, No. 2, pp. 110–115, 2017.

[9] M. Mackay, T. Baker, and A. Al-Yasiri, "Security-oriented cloud computing platform for critical infrastructures," *Computer Law and Security Review*, vol. 28, pp. 679–686, 2012.
[10] G. Manogaran, C. Thota, and M. V. Kumar, "Metaclouddatastorage architecture for big data security in cloud computing," *Procedia Computer Science*, vol. 87, pp. 129–133, 2016.
[11] Y. Ming and Y. Wang,"On the security of three public auditing schemes in cloud computing," *International Journal of Network Security* , vol. 17, No. 6 pp. 795–802, 2015.
[12] A. Mosa, H. M. El-Bakry, S. M. Abd El-Razek, S. Q. Hasan, "A proposed E-government framework based on cloud service architecture," *International Journal of Electronics and Information Engineering*, vol. 5, no. 2, pp. 93–104, 2016.
[13] R. V. Rao and K. Selvamani, "Data security challenges and its solutions in cloud computing," *Procedia Computer Science*, vol. 48, pp. 204–209, 2015.
[14] H. Rasheed, "Data and infrastructure security auditing in cloud computing environments," *International Journal of Information Management*, vol. 34, no. 3, pp. 364–368, 2014.
[15] C. Rong, S. T. Nguyen, and M. G. Jaatun, "Beyond lightning: A survey on security challenges in cloud computing," *Computers and Electrical Engineering*, vol. 39, pp. 47–54, 2013.
[16] A. P. Singh and S. K. Pasupuleti, "Optimized public auditing and data dynamics for data storage security in cloud computing," *Procedia Computer Science*, vol. 93, pp. 751–759, 2016.
[17] S. Singh, Y. S. Jeong, and J. H. Park, "A survey on cloud computing security: Issues, threats, and solutions," *Journal of Network and Computer Applications*, vol. 75, pp. 200–222, 2016.
[18] R. Trapero, J. Modic, M. Stopar, A. Taha, and N. Suri, "A novel approach to manage cloud security SLA incidents," *Future Generation Computer Systems*, vol. 72, pp. 193-205, 2017.
[19] N. Vurukonda and B. T. Rao, "A study on data storage security issues in cloud computing," *Procedia Computer Science*, vol. 92, pp. 128–135, 2016.
[20] L. Wei, H. Zhu, Z. Cao, X. Dong, W. Jia, Y. Chen, and A. V. Vasilakos, "Security and privacy for storage and computation in cloud computing," *Information Sciences*, vol. 258, pp. 371–386, 2014.
[21] D. Wu, B. Yang, and R. Wang, "Scalable privacy-preserving big data aggregation mechanism," *Digital Communications and Networks*, vol. 2, no. 3, pp. 122–129, 2016.

Biography

Yunfa Li is a Ph.D.and associate professor in school of Computer Science and Technology at Hangzhou Dianzi University. His research interests include cloud computing, cluster computing, grid computing, big data and system security. Contact him at yunfali@hdu.edu.cn.

Yangyang Shen is a postgraduate in school of Computer Science and Technology at Hangzhou Dianzi University. His research interests include cloud computing, big data and system security. Contact him at 2644294165@qq.com.

Mingyi Li is a postgraduate in school of Computer Science and Technology at Hangzhou Dianzi University. His research interests include cloud computing, big data and system security. Contact him at 952921628@qq.com.

Traceable Certificateless Ring Signature Scheme For No Full Anonymous Applications

Ke Gu[1,2], LinYu Wang[1], Na Wu[1] and NianDong Liao[1]
(Corresponding author: Ke Gu)
School of Computer and Communication Engineering, Changsha University of Science and Technology[1]
Wangjiali Rd, Tianxin district, Changsha, Hunan Province 410114, China
Hunan Provincial Key Laboratory of Intelligent Processing of Big Data on Transportation,
Changsha University of Science and Technology[2]
(Email: gk4572@163.com)
(Received Jan. 26, 2017; revised and accepted June 25, 2017)

Abstract

With the rapid development of identity-based cryptography, several traceable (or linkable) identity-based ring signature (TIBRS) schemes have been proposed. Compared with ring signature based on public key cryptography, TIBRS can simplify public key management and be used for more applications. However, identity-based cryptography still has the problem of private key management and few traceable ring signature schemes are constructed in the standard model. In this paper, we present a fully traceable certificateless ring signature (TCRS) scheme in the standard model, which has a security reduction to the computational Diffie-Hellman (CDH) assumption.

Keywords: Certificateless Cryptography; Ring Signature; Standard Model; Traceability

1 Introduction

Ring signature [1, 12, 22, 24, 42, 49, 51] allows ring member to hide his identifying information to a ring when ring member signs any message, thus ring signature only reveals the fact that a message was signed by possible one of ring members (a list of possible signers). Ring signature is also called as a special group signature [20]. However, compared with group signature, ring signature has more advantages: the group (ring) must not be constructed by a group manager, who can revoke the anonymity of any signer or identify the real group signer; additionally, because a list of possible signers must be constructed to form a group, some intricate problems need to be solved in a group signature scheme, such as joining the new members and the revocation of group members. Although ring signature can provide more flexibility and full anonymity, it is vulnerable to keep the signers from abusing their signing rights. Namely, it is infeasible for the verifier to determine whether the signatures are generated by the same signer on the same event. Thus, in a practical ring signature scheme, the third trusted authority or the verifier must be able to know who signs the messages on the same event many times and the verifier can not accept the signatures generated by the same signer on the same event [2, 10, 15, 33, 35, 39].

Traceable ring signature[1] [27] is a ring signature that restricts abusing anonymity. Unlike group signature has too strong a traceability characteristic and ring signature has too strong an anonymity characteristic, traceable ring signature has the balance characteristic of anonymity and traceability. Namely, traceable ring signature provides restricted anonymity and traceability. In a traceable ring signature scheme, traceable ring signature can provide full anonymity for the responsible or honest signer when the singer signs any message and provide traceability for the verifier (or the third trusted authority) to determine whether the signatures are generated by the same signer on the same event when the irresponsible signer abuses anonymity in some applications. In order to achieve this requirement of traceable ring signature, we need to consider the two notions "one-more unforgeability" and "double-spending traceability" [18, 19, 27] in the context of ring signature, which originate from blind signature. First, any user can not generate a "one-more" new signature after he obtained a signature from the original signer. Second, if an irresponsible user signs any message twice on the same event, the signatures generated by the user can be traced to reveal the identity of the signer [14, 40]. In the second notion, a responsible user can be anonymously protected. Obviously, traceable ring signature can provide more practicality because of its restricted anonymity in many no full anonymous applications.

Currently ring signatures are used in many different applications, such as whistle blowing [42], anonymous au-

[1]This notion is closely related to linkable ring signature in [5,35–37].

thentication for ad-hoc network [35], e-voting [21] and e-cash [45], non-interactive deniable authentication [44] and multi designated verifiers signature [34], *etc.* Because ring signature is not linkable, no one can determine whether two ring signatures are generated by the same signer. Thus, it exists high risk that ring signatures are used in e-voting and e-cash. For example, if a user signs a message twice for double votes in anonymous e-voting, no one can find the two signatures are linkable so as to detect the irregularity. Obviously, traceable ring signature is suitable for the kind of applications, because it can find the two signatures are linkable. There also are other applications for traceable ring signature. In the "off-line" anonymous e-cash systems, a user is permitted to anonymously signs a message once during one cash transaction, thus traceable ring signature is a natural choice for this application [27]. Damgard *et al.* [23] proposed an unclonable group identification without the group manager, traceable ring signature is also suitable for this application because of not employing the group manager and its balance of anonymity and traceability.

In public key cryptography, the management of public keys is a critical problem. For example, certificate authority (CA) generates a digital certificate, which assures that public key belongs to corresponding user [38]. Thus, in a ring signature scheme based on public key cryptography, because a list (ring) of public keys is corresponding to ring member's private keys (signing keys), the management cost of public keys is proportional to the number of ring members. Additionally, in the ring signature schemes based on public key cryptography, the proposed schemes also suffers from other drawbacks such as verification and revocation of certificates. Obviously, removing public key certificates can simplify the procedure of joining and revocation of ring member. Identity-based cryptography is another cryptographic primitive. In identity-based cryptography, a user's public key is obtained from his/her public identity, such as name, IP address or email address, *etc.* Thus, the user's private key is distributed from a private key generator (PKG). The main target of application of identity-based cryptography is to simplify public key management and remove public key certificates. However, identity-based cryptography still has the problem of private key management. For example, the private key generator may be not fully trusted or be corrupted, so identity-based cryptography has a certain risk in practice. Al-Riyami and Paterson [3] proposed the certificateless public key cryptography, which not only solves the problem of private key management, but also removes public key certificates. Therefore, compared with ring signatures based on public key cryptography and identity-based cryptography, certificateless ring signature [17, 29] can lessen the risk of private key management and the suffering of joining and revocation of ring member.

In this paper, we present a traceable certificateless ring signature scheme in the standard model, which has the properties of anonymity and traceability with enough security.

2 Related Works

Liu *et al.* [35] first proposed the notion of linkable ring signature. In their scheme, if an irresponsible user anonymously signs any message twice on the same event, the two signatures generated by the user can be linked. Base on this notion, some similar schemes were proposed in [4, 35–37, 45, 46]. In [35, 36], the proposed schemes cannot resist the attack that an irresponsible signer forges the signature of a honest signer so as to make the honest signer accused of "double-signing". In [4, 46], the proposed schemes overcome this weakness, but the security conditions are more complicated. In [45], Tsang *et al.* proposed a short linkable ring signature scheme, which is based on the group identification scheme from [25]. Their scheme provides weak traceability, namely it can only detect the linkable ring signatures. In [46], Tsang *et al.* proposed a separable linkable threshold ring signature scheme, where the threshold setting is to restrict abusing signing. However, their scheme is complicated. In [53], Liu *et al.* proposed a revocable ring signature scheme, which supports that any ring member may revoke the anonymity of the real signer when the ring signature is proved to be argumentative. Their scheme provides that all the ring members can reveal the identity of the real signer of any ring signature generated on behalf of their ring. In 2007 and 2011, Fujisaki *et al.* [27, 28] proposed two traceable ring signature schemes and a security model of traceable ring signature was formally proposed. In their scheme, if two signatures are linked, the identity of this signer will be revealed. In other words, the anonymity of the signer will be revoked if and only if the signer generates two ring signatures on the same event. Compared with revocable ring signature [53], traceable ring signature needs the condition of revoking anonymity that the same signer generates two ring signatures on the same event. However, the two secure schemes [27, 28] are based on public key cryptography. With the rapid development of identity-based cryptography [11, 13, 41, 47], many researchers proposed many identity-based signature (IBS) schemes in the random oracle model or standard model [9, 16, 30, 41]. Also, with these identity-based signature schemes, a lot of variants, such as the identity-based proxy signature schemes [43, 48, 50], the identity-based ring signature schemes [5–8, 43], the identity-based group signature schemes [26, 31], *etc.* have also been proposed. In 2006, Au *et al.* [6] proposed a constant size identity-based linkable and revocable-iff-linked ring signature. However, their scheme was later proved to be insecure [32]. In 2012, Au *et al.* [5] proposed a new identity-based event-oriented linkable ring signature scheme with an option as revocable-iff-linked. With this option, if a user generates two linkable ring signatures in the same event, everyone can compute his identity from these two signatures. In the Au *et al.*'s frame, they consider the PKG system is partially trusted, which is similar to certificateless public key cryptography. However, their scheme is constructed in the random oracle model.

3 Preliminaries

3.1 Bilinear Maps

Let $\mathbb{G}_1$ and $\mathbb{G}_2$ be groups of prime order q and g be a generator of $\mathbb{G}_1$. We say $\mathbb{G}_2$ has an admissible bilinear map, $e : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2$ if the following two conditions hold. The map is bilinear; for all a, b, we have $e\left(g^a, g^b\right) = e(g,g)^{a \cdot b}$. The map is non-degenerate; we must have that $e\left(g, g\right) \neq 1$.

3.2 Computational Diffie-Hellman Assumption

Definition 1. *Computational Diffie-Hellman (CDH) Problem: Let $\mathbb{G}_1$ be a group of prime order q and g be a generator of $\mathbb{G}_1$; for all $(g, g^a, g^b) \in \mathbb{G}_1$, with $a, b \in \mathbb{Z}_q$, the CDH problem is to compute $g^{a \cdot b}$.*

Definition 2. *The $(\hbar, \varepsilon)$-CDH assumption holds if no $\hbar$-time algorithm can solve the CDH problem with probability at least ε.*

4 A Framework for TCRS

In the section, we present a formal definition of TCRS. Let $\mathbb{A}$ be universe of possible identities, we set $ID \subseteq \mathbb{A}$ as the identity of user.

Definition 3. *Traceable Certificateless Ring Signature Scheme:* Let *TCRS*=(System-Setup, Generate-Key, Sign, Verify, Trace-User) be a traceable certificateless ring signature scheme on $\mathbb{A}$, where the algorithm Generate-Key includes four sub-procedures[2]. In *TCRS*, all algorithms are described as follows:

1) *System-Setup: The randomized algorithm run by key generate center (KGC) inputs a security parameter 1^k and then outputs all system parameters $TCRK$ and a system private key spk on the security parameter 1^k.*

2) *Generate-Key: The randomized algorithm run by key generate center (or user) inputs ($TCRK$, spk, $ID_i \subseteq \mathbb{A}$) and then the following steps are finished:*

   - *Generate-Partial Key: The algorithm run by key generate center outputs a user's partial private key psk_{ID_i} to a ring member, where ID_i is the identity of the ring member with $i \in \{1, 2.....n\}$ (n is the number of the ring members in a ring).*
   - *Set-Secret: The algorithm run by the ring member outputs the corresponding secret sx_{ID_i} according to ID_i.*
   - *Generate-Signing Key: The algorithm run by the ring member outputs the corresponding signing (private) key sk_{ID_i} according to psk_{ID_i} and sx_{ID_i}.*
   - *Generate-Public Key: The algorithm run by the ring member outputs and publishes the corresponding public key pk_{ID_i}.*

3) *Sign: The randomized algorithm is a standard traceable certificateless ring signature algorithm. A ring member needs to sign a message $\mathfrak{M} \in \{0,1\}^*$ on an event identifier $\mathfrak{E} \in \{0,1\}^*$. The algorithm run by the ring member with the identity ID_i inputs ($TCRK$, sk_{ID_i}, RL_ID, RL_PK, $\mathfrak{M}$, $\mathfrak{E}$) and then outputs a signature σ, where RL_ID is an identity list including all identities of the ring members belong to this ring, RL_PK is a public key list including all public keys of the ring members belong to this ring, $\sigma \in \{0,1\}^* \cup \{\perp\}$, sk_{ID_i} is the signing key of the ring member with $i \in \{1, 2.....n\}$.*

4) *Verify: The signature verifiers verify a standard traceable certificateless ring signature σ. The deterministic algorithm run by a signature verifier inputs ($TCRK$, RL_ID, RL_PK, $\mathfrak{M}$, $\mathfrak{E}$, σ) and then outputs the boolean value, accept or reject.*

5) *Trace-User: The trusted authority traces a real ring member (signer) by two traceable certificateless ring signatures σ_1 on $\mathfrak{M}_1$ and σ_2 on $\mathfrak{M}_2$. The deterministic algorithm run by the trusted authority inputs ($TCRK$, RL_ID, RL_PK, $\{\mathfrak{M}_1, \sigma_1\}$, $\{\mathfrak{M}_2, \sigma_2\}$, $\mathfrak{E}$) and then outputs one of the following results: "the identity ID of the real signer", or "Independent" or "Linked", where $ID \in RL\_ID$.*

5 Traceable Certificateless Ring Signature Scheme

In the section, we show a traceable certificateless ring signature scheme in the standard model under our framework for TCRS. Let TCRS=(*System-Setup*, *Generate-Key*, *Sign*, *Verify*, *Trace-User*) be a traceable certificateless ring signature scheme. In TCRS, all algorithms are described as follows.

1) TCRS.*System-Setup*: The algorithm run by the KGC system inputs a security parameter 1^k. Additionally, let $\mathbb{G}_1$ and $\mathbb{G}_2$ be groups of prime order q and g be a generator of $\mathbb{G}_1$ and let $e : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2$ denote the bilinear map. The size of the group is determined by the security parameter and we set $\mathbb{A} \subseteq \mathbb{Z}_q$ as the universe of identities. And one hash function, $H : \{0,1\}^* \to \mathbb{Z}_{1^k \cdot q}$ can be defined and used to generate any integer value in $\mathbb{Z}_{1^k \cdot q}$ (where 1^k represents the corresponding decimal number).

   Then the system parameters are generated as follows for a ring system setup. The algorithm chooses a

[2] In certificateless public key cryptography, the algorithm *Generate-Key* is divided to four algorithms.

random $a \in \mathbb{Z}_q$ and then sets $g_1 = g^a$. Eight group elements g_2, ϑ, ψ, ϖ, μ, τ, χ and $\kappa \in \mathbb{G}_1$ are randomly chosen. Finally, the algorithm outputs the public parameters $TCRK$=($\mathbb{G}_1$, $\mathbb{G}_2$, e, g, g_1, g_2, ϑ, ψ, ϖ, μ, τ, χ, κ), where $spk = g_2^a$ is seen as a master private key.

2) TCRS.Generate-Key: The algorithm run by the KGC system generates user's signing key with respect to the identity of ring member when user joins ring. The algorithm inputs ($TCRK$, spk, $ID \subseteq \mathbb{A}$), where ID is the identity of a ring member and then the following steps are finished:

- Generate-Partial Key: The algorithm run by the KGC system randomly chooses $r_1, r_L \in \mathbb{Z}_q$, computes $x_0 = g_2^a \cdot \vartheta^{r_1 \cdot H(ID)} \cdot \psi^{r_1} \cdot \varpi^{r_L}$, $x_1 = g^{r_1}$, $sx_L = g^{r_L}$. The algorithm outputs a partial private key $psk_{\{ID\}} = \{x_0, x_1, sx_L\}$ to the ring member and publishes a new identity ring RL_ID, where sx_L is the traced ring secret for the ring member, RL_ID is an identity list including all identities of the ring members belong to this ring and $ID \in RL\_ID$.

  Remark 1. *Every ring member may verify his partial private key by the following equation:*

$$\begin{aligned} e(x_0, g) &= e(g_1, g_2) \cdot e(\vartheta, x_1^{H(ID)}) \cdot e(\psi, x_1) \\ &\quad \cdot e(\varpi, sx_L). \end{aligned}$$

- Set-Secret: The algorithm run by the corresponding ring member randomly chooses $r_2 \in \mathbb{Z}_q$, computes the member secret $sx_{\{ID\}} = \vartheta^{r_2 \cdot H(ID)} \cdot \psi^{r_2}$.
- Generate-Signing Key: The algorithm run by the corresponding ring member computes $x_2 = x_0 \cdot sx_{\{ID\}} = g_2^a \cdot \vartheta^{r_1 \cdot H(ID)} \cdot \psi^{r_1} \cdot \varpi^{r_L} \cdot \vartheta^{r_2 \cdot H(ID)} \cdot \psi^{r_2} = g_2^a \cdot \vartheta^{(r_1+r_2) \cdot H(ID)} \cdot \psi^{r_1+r_2} \cdot \varpi^{r_L}$ and then outputs the signing key $sk_{\{ID\}} = \{x_1, x_2, sx_L\}$.
- Generate-Public Key: The algorithm run by the corresponding ring member outputs and publishes the public key $pk_{\{ID\}} = g^{r_2}$, which is added to the public key ring RL_PK, where RL_PK is a public key list including all public keys of the ring members belong to this ring and $pk_{\{ID\}} \in RL\_PK$.

3) TCRS Sign: A ring member with the identity ID needs to sign a message $\mathfrak{M} \in \{0,1\}^*$ on an event identifier $\mathfrak{E} \in \{0,1\}^*$. The algorithm run by the ring member inputs ($TCRK$, $sk_{\{ID\}}$, RL_ID, RL_PK, $\mathfrak{M}$, $\mathfrak{E}$) and then randomly chooses $r_3, r_4, r_5 \in \mathbb{Z}_q$, computes

$$\begin{aligned} \sigma_0 &= x_2 \cdot \vartheta^{r_3 \cdot H(ID)} \cdot \psi^{r_3} \cdot \varpi^{r_3} \cdot \mu^{r_4 \cdot H(RL\_ID \| RL\_PK)} \cdot \\ &\quad \tau^{r_4} \cdot \chi^{r_5 \cdot H(\mathfrak{M} \| \mathfrak{E})} \cdot \kappa^{r_5} \\ &= g_2^a \cdot \vartheta^{(r_1+r_2+r_3) \cdot H(ID)} \cdot \psi^{r_1+r_2+r_3} \cdot \varpi^{r_L+r_3} \cdot \\ &\quad \mu^{r_4 \cdot H(RL\_ID \| RL\_PK)} \cdot \tau^{r_4} \cdot \chi^{r_5 \cdot H(\mathfrak{M} \| \mathfrak{E})} \cdot \kappa^{r_5}, \\ \sigma_1 &= e(\vartheta^{H(ID)} \cdot \psi, x_1 \cdot pk_{\{ID\}}) \cdot e(\vartheta^{r_3 \cdot H(ID)} \cdot \psi^{r_3}, g) \\ &= e(\vartheta^{H(ID)} \cdot \psi, g^{r_1+r_2}) \cdot e(\vartheta^{r_3 \cdot H(ID)} \cdot \psi^{r_3}, g) \\ &= e(\vartheta^{(r_1+r_2+r_3) \cdot H(ID)} \cdot \psi^{r_1+r_2+r_3}, g), \\ \sigma_2 &= sx_L \cdot g^{r_3} = g^{r_L+r_3}, \\ \sigma_3 &= g^{r_4}, \\ \sigma_4 &= g^{r_5}. \end{aligned}$$

Finally, the algorithm outputs a signature $\Phi = \{\sigma_0, \sigma_1, \sigma_2, \sigma_3, \sigma_4\}$.

4) TCRS.Verify: The signature verifiers verify a standard traceable certificateless ring signature Φ. The algorithm run by a signature verifier inputs ($TCRK$, RL_ID, RL_PK, $\mathfrak{M}$, $\mathfrak{E}$, Φ) and then the following computation is finished:

$$\begin{aligned} e(\sigma_0, g) &= e(g_1, g_2) \cdot \sigma_1 \\ &\quad \cdot e(\varpi, \sigma_2) \cdot e(\mu^{H(RL\_ID \| RL\_PK)} \cdot \tau, \sigma_3) \\ &\quad \cdot e(\chi^{H(\mathfrak{M} \| \mathfrak{E})} \cdot \kappa, \sigma_4). \end{aligned}$$

If the above equation is correct, then the algorithm outputs the boolean value *accept*, otherwise the algorithm outputs the boolean value *reject*.

5) TCRS.Trace-User: The trusted authority traces a ring member (signer) by two traceable certificateless ring signatures Φ_1 on $\mathfrak{M}_1$ and Φ_2 on $\mathfrak{M}_2$ when the signatures need to be arbitrated. The algorithm run by the trusted authority inputs ($TCRK$, RL_ID, RL_PK, $\{\mathfrak{M}_1, \Phi_1\}$, $\{\mathfrak{M}_2, \Phi_2\}$, $\mathfrak{E}$), where the trusted authority may get x_1 and sx_L from the KGC or the ring members[3] and then the following steps are finished:

a. For any potential identity $ID_1 \in RL\_ID$ and the tuple $\{\mathfrak{M}_1, \Phi_1\}$, the algorithm computes the equation:

$$e(\vartheta^{H(ID_1)} \cdot \psi, x_1 \cdot pk_{\{ID\}} \cdot \tfrac{\sigma_2}{sx_L}) = \frac{e(\sigma_0, g)}{e(g_1, g_2) \cdot e(\varpi, \sigma_2) \cdot e(\mu^{H(RL\_ID \| RL\_PK)} \cdot \tau, \sigma_3) \cdot e(\chi^{H(\mathfrak{M}_1 \| \mathfrak{E})} \cdot \kappa, \sigma_4)}.$$

If the above equation is correct, then the algorithm securely records the identity ID_1 of the real signer, otherwise if the algorithm does not find the corresponding identity, the algorithm aborts; similarly, the same computation is finished for any potential identity $ID_2 \in RL\_ID$ and the tuple $\{\mathfrak{M}_2, \Phi_2\}$ and then the algorithm securely records the identity ID_2 of the real signer, otherwise the algorithm aborts.

b. The algorithm outputs the following results according to the comparisons:

- *Result*="*Independent*", if $ID_1 \neq ID_2$;
- *Result*="*Linked*", else if $\mathfrak{M}_1 = \mathfrak{M}_2$;
- *Result*="ID_1", otherwise.

[3] This setting does not break the security of the whole scheme according to the Paterson *et al.*'s signature scheme [41].

6 Analysis of the Proposed Scheme

6.1 Correctness

In the proposed scheme, the traceable certificateless ring signature is $\Phi = \{\sigma_0, \sigma_1, \sigma_2, \sigma_3, \sigma_4\}$, where

$$\begin{aligned}
\sigma_0 &= x_2 \cdot \vartheta^{r_3 \cdot H(ID)} \cdot \psi^{r_3} \cdot \varpi^{r_3} \cdot \mu^{r_4 \cdot H(RL_ID \| RL_PK)} \\
&\quad \cdot \tau^{r_4} \cdot \chi^{r_5 \cdot H(\mathfrak{M} \| \mathfrak{E})} \cdot \kappa^{r_5} \\
&= g_2^a \cdot \vartheta^{(r_1+r_2+r_3) \cdot H(ID)} \cdot \psi^{r_1+r_2+r_3} \cdot \varpi^{r_L+r_3} \\
&\quad \cdot \mu^{r_4 \cdot H(RL_ID \| RL_PK)} \cdot \tau^{r_4} \cdot \chi^{r_5 \cdot H(\mathfrak{M} \| \mathfrak{E})} \cdot \kappa^{r_5}, \\
\sigma_1 &= e(\vartheta^{H(ID)} \cdot \psi, x_1 \cdot pk_{\{ID\}}) \cdot e(\vartheta^{r_3 \cdot H(ID)} \cdot \psi^{r_3}, g) \\
&= e(\vartheta^{H(ID)} \cdot \psi, g^{r_1+r_2}) \cdot e(\vartheta^{r_3 \cdot H(ID)} \cdot \psi^{r_3}, g) \\
&= e(\vartheta^{(r_1+r_2+r_3) \cdot H(ID)} \cdot \psi^{r_1+r_2+r_3}, g), \\
\sigma_2 &= sx_L \cdot g^{r_3} = g^{r_L+r_3}, \\
\sigma_3 &= g^{r_4}, \\
\sigma_4 &= g^{r_5}.
\end{aligned}$$

So, we have that

$$\begin{aligned}
e(\sigma_0, g) &= e(g_2^a \cdot \vartheta^{(r_1+r_2+r_3) \cdot H(ID)} \cdot \psi^{r_1+r_2+r_3} \\
&\quad \cdot \varpi^{r_L+r_3} \cdot \mu^{r_4 \cdot H(RL_ID \| RL_PK)} \cdot \tau^{r_4} \\
&\quad \cdot \chi^{r_5 \cdot H(\mathfrak{M} \| \mathfrak{E})} \cdot \kappa^{r_5}, g) \\
&= e(g_2^a, g) \cdot e(\vartheta^{(r_1+r_2+r_3) \cdot H(ID)} \cdot \psi^{r_1+r_2+r_3}, g) \\
&\quad \cdot e(\varpi^{r_L+r_3}, g) \cdot e(\mu^{r_4 \cdot H(RL_ID \| RL_PK)} \cdot \tau^{r_4}, g) \\
&\quad \cdot e(\chi^{r_5 \cdot H(\mathfrak{M} \| \mathfrak{E})} \cdot \kappa^{r_5}, g) \\
&= e(g_1, g_2) \cdot \sigma_1 \cdot e(\varpi, \sigma_2) \cdot e(\mu^{H(RL_ID \| RL_PK)} \\
&\quad \cdot \tau, \sigma_3) \cdot e(\chi^{H(\mathfrak{M} \| \mathfrak{E})} \cdot \kappa, \sigma_4).
\end{aligned}$$

6.2 Efficiency

In the proposed scheme, $\Phi = \{\sigma_0, \sigma_1, \sigma_2, \sigma_3, \sigma_4\}$, where

$$\begin{aligned}
\sigma_0 &= x_2 \cdot \vartheta^{r_3 \cdot H(ID)} \cdot \psi^{r_3} \cdot \varpi^{r_3} \cdot \mu^{r_4 \cdot H(RL_ID \| RL_PK)} \\
&\quad \cdot \tau^{r_4} \cdot \chi^{r_5 \cdot H(\mathfrak{M} \| \mathfrak{E})} \cdot \kappa^{r_5}, \\
\sigma_1 &= e(\vartheta^{H(ID)} \cdot \psi, x_1 \cdot pk_{\{ID\}}) \cdot e(\vartheta^{r_3 \cdot H(ID)} \cdot \psi^{r_3}, g), \\
\sigma_2 &= sx_L \cdot g^{r_3}, \quad \sigma_3 = g^{r_4}, \quad \sigma_4 = g^{r_5}.
\end{aligned}$$

Thus, the length of signature is $4 \cdot |\mathbb{G}_1| + |\mathbb{G}_2|$, where $|\mathbb{G}_1|$ is the size of element in $\mathbb{G}_1$ and $|\mathbb{G}_2|$ is the size of element in $\mathbb{G}_2$. Additionally, because $x_2 \cdot \vartheta^{r_3 \cdot H(ID)} \cdot \psi^{r_3} \cdot \varpi^{r_3} \cdot \mu^{r_4 \cdot H(RL\_ID \| RL\_PK)} \cdot \tau^{r_4} \cdot \kappa^{r_5}$, χ^{r_5} in $\chi^{r_5 \cdot H(\mathfrak{M} \| \mathfrak{E})}$, σ_1, σ_2, σ_3 and σ_4 may be precomputed and we assume that the time for integer multiplication and hash computation can be ignored, signing a message for a traceable certificateless ring signature only needs to compute at most 1 exponentiation in $\mathbb{G}_1$ and 1 multiplication in $\mathbb{G}_1$. Also, in the following equation

$$\begin{aligned}
e(\sigma_0, g) = e(g_1, g_2) \cdot \sigma_1 \cdot e(\varpi, \sigma_2) \cdot e(\mu^{H(RL_ID \| RL_PK)} \cdot \\
\tau, \sigma_3) \cdot e(\chi^{H(\mathfrak{M} \| \mathfrak{E})} \cdot \kappa, \sigma_4),
\end{aligned}$$

because the value $e(g_1, g_2)$ can be precomputed and cached, verification requires 4 pairing computations, 2 exponentiations in $\mathbb{G}_1$, 2 multiplications in $\mathbb{G}_1$ and 4 multiplications in $\mathbb{G}_2$.

In this paper, we compare the proposed scheme (the scheme of Section 5) with other traceable (or linkable) ring signature schemes proposed by [5, 27, 28, 36, 52]. Table 1 shows the comparisons of the traceable or linkable ring signature schemes. Compared with other schemes, our scheme is certificateless and constructed in the standard model and has constant signature size in the comparison of the performance.

6.3 Security

In the section, we show the proposed scheme (the scheme of Section 5) has a security reduction to the CDH assumption and the TCRS unforgeability (against *linkability attacks* and *exculpability attacks*) under the adaptive chosen message and identity attacks and has the TCRS anonymity. Our proofs of the following theorems are based on the security models of [5, 27][4].

Theorem 1. *The scheme of Section 5 is ($\hbar$, ε, q_g, q_s)-unforgeable, assuming that the ($\hbar'$, ε')-CDH assumption holds in $\mathbb{G}_1$, where*

$$\varepsilon' = [(1 - \tfrac{q_g}{q}) \cdot (1 - \tfrac{q_s}{q})^2 \cdot \tfrac{\varepsilon}{q^2}] \parallel [(1 - \tfrac{q_g}{q}) \cdot (1 - \tfrac{q_s}{q})^2 \cdot \varepsilon],$$

$$\hbar' = \max\{\hbar + O(q_g \cdot (10 \cdot C_{exp} + 3 \cdot C_{mul}) + q_s \cdot (15 \cdot C_{exp} + 11 \cdot C_{mul})), \hbar + O(q_g \cdot (3 \cdot C_{exp} + C_{mul}) + q_s \cdot (12 \cdot C_{exp} + 7 \cdot C_{mul}))\},$$

and q_g is the maximal number of "Generate-Key" oracle queries, q_s is the maximal number of "Sign" oracle queries, C_{mul} and C_{exp} are respectively the time for a multiplication and an exponentiation in $\mathbb{G}_1$.

Proof. The procedure of the whole proof is divided to two following parts for two types of attack. □

Type I:

Let **TCRS** be a traceable certificateless ring signature scheme of Section 5. Additionally, let $\mathcal{A}$ be an ($\hbar$, ε, q_g, q_s)-adversary attacking **TCRS**. From the adversary $\mathcal{A}$, we construct an algorithm $\mathcal{B}$, for (g, g^a, g^b)$\in \mathbb{G}_1$, the algorithm $\mathcal{B}$ is able to use $\mathcal{A}$ to compute $g^{a \cdot b}$. Thus, we assume the algorithm $\mathcal{B}$ can solve the CDH with probability at least ε' and in time at most $\hbar'$, contradicting the ($\hbar'$, ε')-CDH assumption. Such a simulation may be created in the following way.

Setup: The KGC system inputs a security parameter 1^k. Additionally, let $\mathbb{G}_1$ and $\mathbb{G}_2$ be groups of prime order q and g be a generator of $\mathbb{G}_1$ and let $e : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2$ denote the bilinear map. The size of the group is determined by the security parameter and we set $\mathbb{A} \subseteq \mathbb{Z}_q$ as the universe of identities. One hash function, $H : \{0,1\}^* \to \mathbb{Z}_{1^k \cdot q}$ can be defined and used to generate any integer value in $\mathbb{Z}_{1^k \cdot q}$ (where 1^k represents the corresponding decimal number).

Then the system parameters are generated as follows. The algorithm sets $g_1 = g^a$ and $g_2 = g^b$ with $a, b \in$

[4] As the proofs of Theorem 2, Theorem 3 and Theorem 4 are similar to the proof of Theorem 1, we omit the similar proofs in this paper.

Table 1: Comparisons of the six schemes

	Signature Size	Cryptography	Traceability	Linking Cost	Model
Scheme [36]	$O(n)$	Public Key	No	$O(1)$	random oracle model
Scheme [52]	$O(n)$	Public Key	No	$O(1)$	random oracle model
Scheme [28]	$O(\sqrt{n})$	Public Key	Yes	$O(n \cdot \log n)$	standard model
Scheme [27]	$O(n)$	Public Key	Yes	$O(n)$	random oracle model
Scheme [5]	$O(1)$	Identity-Based	Yes	$O(1)$	random oracle model
Our Scheme	$O(1)$	Certificateless	Yes	$O(n)$	standard model

$\mathbb{Z}_q$ ($\mathcal{B}$ doesn't know a and b). Also, the algorithm chooses ℓ, ∂, ν, λ, η, α and $\pi \in \mathbb{Z}_q$ and then sets $\vartheta = g_2^{\ell} \cdot g$, $\psi = g^{\partial}$, $\mu = g^{\nu}$, $\tau = g^{\lambda}$, $\chi = g_2^{\alpha} \cdot g$, $\kappa = g^{\pi}$ and $\varpi = g^{\eta}$. Finally, the system outputs the public parameters $TCRK$=($\mathbb{G}_1$, $\mathbb{G}_2$, e, g, g_1, g_2, ϑ, ψ, μ, τ, χ, κ, ϖ).

Queries: When running the adversary $\mathcal{A}$, the relevant queries can occur. The algorithm $\mathcal{B}$ answers these in the following way:

- Generate-Key Queries:
  Given the public parameters $TCRK$ and the identity ID of the ring member ($ID \in RL\_ID$ where RL_ID is an identity list), the algorithm $\mathcal{B}$ can construct a private key of the ring member u by the following computation (ID is the identity of u):
  - Generate-Partial Key:
    The algorithm chooses random $r_1, r_L \in \mathbb{Z}_q$ and computes $x_0 = g_1^{-\frac{1}{\ell}} \cdot \vartheta^{r_1} \cdot g_1^{-\frac{\partial}{\ell} \cdot \frac{1}{H(ID)}} \cdot \psi^{\frac{r_1}{H(ID)}} \cdot \varpi^{r_L}$, $x_1 = (g_1^{-\frac{1}{\ell}} \cdot g^{r_1})^{\frac{1}{H(ID)}}$, $sx_L = g^{r_L}$ and then generates a partial private key $psk_{\{ID\}} = \{x_0, x_1, sx_L\}$.
    Remark 2. *To the correctness of $psk_{\{ID\}}$, $psk_{\{ID\}}$ may be changed as follows:*

$$\begin{aligned}
x_0 &= g_1^{-\frac{1}{\ell}} \cdot \vartheta^{r_1} \cdot g_1^{-\frac{\partial}{\ell} \cdot \frac{1}{H(ID)}} \cdot \psi^{\frac{r_1}{H(ID)}} \cdot \varpi^{r_L} \\
&= g_2^{a} \cdot g_2^{-a} \cdot g_1^{-\frac{1}{\ell}} \cdot \vartheta^{r_1} \cdot g_1^{-\frac{\partial}{\ell} \cdot \frac{1}{H(ID)}} \cdot \psi^{\frac{r_1}{H(ID)}} \cdot \varpi^{r_L} \\
&= g_2^{a} \cdot (g_2^{\ell} \cdot g)^{-\frac{a}{\ell}} \cdot \vartheta^{r_1} \cdot g^{a \cdot (-\frac{\partial}{\ell}) \cdot \frac{1}{H(ID)}} \cdot \psi^{\frac{r_1}{H(ID)}} \cdot \varpi^{r_L} \\
&= g_2^{a} \cdot \vartheta^{-\frac{a}{\ell}} \cdot \vartheta^{r_1} \cdot \psi^{-\frac{a}{\ell} \cdot \frac{1}{H(ID)}} \cdot \psi^{\frac{r_1}{H(ID)}} \cdot \varpi^{r_L} \\
&= g_2^{a} \cdot \vartheta^{r_1 - \frac{a}{\ell}} \cdot \psi^{\frac{r_1}{H(ID)} - \frac{a}{\ell} \cdot \frac{1}{H(ID)}} \cdot \varpi^{r_L} \\
&= g_2^{a} \cdot \vartheta^{r_1 - \frac{a}{\ell}} \cdot \psi^{(r_1 - \frac{a}{\ell}) \cdot \frac{1}{H(ID)}} \cdot \varpi^{r_L}, \\
x_1 &= (g_1^{-\frac{1}{\ell}} \cdot g^{r_1})^{\frac{1}{H(ID)}} \\
&= (g^{-\frac{a}{\ell}} \cdot g^{r_1})^{\frac{1}{H(ID)}} \\
&= g^{(r_1 - \frac{a}{\ell}) \cdot \frac{1}{H(ID)}}.
\end{aligned}$$

    Setting $r_1' = (r_1 - \frac{a}{\ell}) \cdot \frac{1}{H(ID)}$, $psk_{\{ID\}} = \{x_0, x_1, sx_L\} = \{g_2^{a} \cdot \vartheta^{r_1' \cdot H(ID)} \cdot \psi^{r_1'} \cdot \varpi^{r_L}, \quad g^{r_1'}, \quad g^{r_L}\}$ is a valid partial private key, where we assure that $\ell \cdot H(ID) \neq 0$ **mod** q.
  - Set-Secret:
    The algorithm randomly chooses $r_0 \in \mathbb{Z}_q$, computes the member secret $sx_{\{ID\}} = \vartheta^{r_0 \cdot H(ID)} \cdot \psi^{r_0}$.
  - Generate-Signing Key:
    The algorithm computes $x_2 = x_0 \cdot sx_{\{ID\}}$ and then outputs the signing key $sk_{\{ID\}} = \{x_1, x_2, sx_L\}$ to $\mathcal{A}$.
  - Generate-Public Key:
    The algorithm outputs the public key $pk_{\{ID\}} = g^{r_0}$, which is added to the public key ring RL_PK, where RL_PK is a public key list including all public keys of the ring members belong to this ring.

- Sign Queries:
  Given the public parameters $TCRK$, the identity list RL_ID ($ID \in RL\_ID$ where ID is the identity of the ring member that belongs to this ring), the public key list RL_PK, the message $\mathfrak{M}$ and the event identifier $\mathfrak{E}$, the algorithm $\mathcal{B}$ chooses random $r_2, r_3, r_4, r_5 \in \mathbb{Z}_q$ and computes

$$\begin{aligned}
\sigma_0 &= g_1^{-\frac{1}{2 \cdot \ell}} \cdot \vartheta^{r_2} \cdot g_1^{-\frac{\partial}{2 \cdot \ell} \cdot \frac{1}{H(ID)}} \cdot \psi^{\frac{r_2}{H(ID)}} \cdot \varpi^{r_3} \cdot \\
&\quad \mu^{r_4 \cdot H(RL_ID \| RL_PK)} \cdot \tau^{r_4} \cdot g_1^{-\frac{1}{2 \cdot \alpha}} \cdot \chi^{r_5} \cdot \\
&\quad g_1^{-\frac{\pi}{2 \cdot \alpha} \cdot \frac{1}{H(\mathfrak{M} \| \mathfrak{E})}} \cdot \kappa^{\frac{r_5}{H(\mathfrak{M} \| \mathfrak{E})}}, \\
\sigma_1' &= (g_1^{-\frac{1}{2 \cdot \ell}} \cdot g^{r_2})^{\frac{1}{H(ID)}}, \\
\sigma_2 &= g^{r_3}, \\
\sigma_3 &= g^{r_4}, \\
\sigma_4 &= (g_1^{-\frac{1}{2 \cdot \alpha}} \cdot g^{r_5})^{\frac{1}{H(\mathfrak{M} \| \mathfrak{E})}}.
\end{aligned}$$

  Finally, the algorithm outputs a forgery $\Phi = \{\sigma_0, \sigma_1', \sigma_2, \sigma_3, \sigma_4\}$ to the adversary $\mathcal{A}$. Where we maximize the adversary's advantage, σ_1' is passed to $\mathcal{A}$.

Remark 3. *To the correctness of Φ, Φ may be*

changed as follows:

$$\begin{aligned}
\sigma_0 &= g_1^{-\frac{1}{2\cdot\ell}}\cdot\vartheta^{r_2}\cdot g_1^{-\frac{\partial}{2\cdot\ell}\cdot\frac{1}{H(ID)}}\cdot \\
&\quad \psi^{\frac{r_2}{H(ID)}}\cdot\varpi^{r_3}\cdot\mu^{r_4\cdot H(RL_ID\|RL_PK)}\cdot \\
&\quad \tau^{r_4}\cdot g_1^{-\frac{1}{2\cdot\alpha}}\cdot\chi^{r_5}\cdot \\
&\quad g_1^{-\frac{\pi}{2\cdot\alpha}\cdot\frac{1}{H(\mathfrak{M}\|\mathfrak{E})}}\cdot\kappa^{\frac{r_5}{H(\mathfrak{M}\|\mathfrak{E})}} \\
&= g_2^{a}\cdot g_2^{-\frac{a}{2}}\cdot g_2^{-\frac{a}{2}}\cdot g_1^{-\frac{1}{2\cdot\ell}}\cdot\vartheta^{r_2}\cdot g_1^{-\frac{\partial}{2\cdot\ell}\cdot\frac{1}{H(ID)}}\cdot \\
&\quad \psi^{\frac{r_2}{H(ID)}}\cdot\varpi^{r_3}\cdot\mu^{r_4\cdot H(RL_ID\|RL_PK)} \\
&\quad \cdot\tau^{r_4}\cdot g_1^{-\frac{1}{2\cdot\alpha}}\cdot\chi^{r_5}\cdot g_1^{-\frac{\pi}{2\cdot\alpha}\cdot\frac{1}{H(\mathfrak{M}\|\mathfrak{E})}}\cdot\kappa^{\frac{r_5}{H(\mathfrak{M}\|\mathfrak{E})}} \\
&\quad g_2^{a}\cdot g_2^{-\frac{a}{2}}\cdot g_1^{-\frac{1}{2\cdot\ell}}\cdot\vartheta^{r_2}\cdot g_1^{-\frac{\partial}{2\cdot\ell}\cdot\frac{1}{H(ID)}}\cdot \\
&\quad \psi^{\frac{r_2}{H(ID)}}\cdot\varpi^{r_3}\cdot\mu^{r_4\cdot H(RL_ID\|RL_PK)}\cdot\tau^{r_4}\cdot \\
&\quad g_2^{-\frac{a}{2}}\cdot g_1^{-\frac{1}{2\cdot\alpha}}\cdot\chi^{r_5}\cdot g_1^{-\frac{\pi}{2\cdot\alpha}\cdot\frac{1}{H(\mathfrak{M}\|\mathfrak{E})}}\cdot \\
&\quad \kappa^{\frac{r_5}{H(\mathfrak{M}\|\mathfrak{E})}}g_2^{a}\cdot(g_2^{\ell}\cdot g)^{-\frac{a}{2\cdot\ell}}\cdot\vartheta^{r_2}\cdot g^{-\frac{a\cdot\partial}{2\cdot\ell}\cdot\frac{1}{H(ID)}}\cdot \\
&\quad \psi^{\frac{r_2}{H(ID)}}\cdot\varpi^{r_3}\cdot\mu^{r_4\cdot H(RL_ID\|RL_PK)}\cdot\tau^{r_4}\cdot \\
&\quad (g_2^{\alpha}\cdot g)^{-\frac{a}{2\cdot\alpha}}\cdot\chi^{r_5}\cdot g^{-\frac{a\cdot\pi}{2\cdot\alpha}\cdot\frac{1}{H(\mathfrak{M}\|\mathfrak{E})}}\cdot\kappa^{\frac{r_5}{H(\mathfrak{M}\|\mathfrak{E})}} \\
&\quad g_2^{a}\cdot\vartheta^{-\frac{a}{2\cdot\ell}}\cdot\vartheta^{r_2}\cdot\psi^{-\frac{a}{2\cdot\ell}\cdot\frac{1}{H(ID)}}\cdot \\
&\quad \psi^{\frac{r_2}{H(ID)}}\cdot\varpi^{r_3}\cdot\mu^{r_4\cdot H(RL_ID\|RL_PK)}\cdot\tau^{r_4}\cdot \\
&\quad \chi^{-\frac{a}{2\cdot\alpha}}\cdot\chi^{r_5}\cdot\kappa^{-\frac{a}{2\cdot\alpha}\cdot\frac{1}{H(\mathfrak{M}\|\mathfrak{E})}}\cdot\kappa^{\frac{r_5}{H(\mathfrak{M}\|\mathfrak{E})}} \\
&\quad g_2^{a}\cdot\vartheta^{r_2-\frac{a}{2\cdot\ell}}\cdot\psi^{(r_2-\frac{a}{2\cdot\ell})\cdot\frac{1}{H(ID)}}\cdot\varpi^{r_3}\cdot \\
&\quad \mu^{r_4\cdot H(RL_ID\|RL_PK)}\cdot\tau^{r_4}\cdot\chi^{r_5-\frac{a}{2\cdot\alpha}}\cdot \\
&\quad \kappa^{(r_5-\frac{a}{2\cdot\alpha})\cdot\frac{1}{H(\mathfrak{M}\|\mathfrak{E})}}, \\
\sigma_1' &= (g_1^{-\frac{1}{2\cdot\ell}}\cdot g^{r_2})^{\frac{1}{H(ID)}} = g^{(r_2-\frac{a}{2\cdot\ell})\cdot\frac{1}{H(ID)}}, \\
\sigma_4 &= (g_1^{-\frac{1}{2\cdot\alpha}}\cdot g^{r_5})^{\frac{1}{H(\mathfrak{M}\|\mathfrak{E})}} = g^{(r_5-\frac{a}{2\cdot\alpha})\cdot\frac{1}{H(\mathfrak{M}\|\mathfrak{E})}}.
\end{aligned}$$

Setting $r_2' = (r_2 - \frac{a}{2\cdot\ell})\cdot\frac{1}{H(ID)}$ and $r_5' = (r_5 - \frac{a}{2\cdot\alpha})\cdot\frac{1}{H(\mathfrak{M}\|\mathfrak{E})}$, $\mathcal{A}$ may get that

$$\begin{aligned}
\sigma_0 &= g_2^{a}\cdot\vartheta^{r_2'\cdot H(ID)}\cdot\psi^{r_2'}\cdot\varpi^{r_3} \\
&\quad \cdot\mu^{r_4\cdot H(RL_ID\|RL_PK)}\cdot\tau^{r_4} \\
&\quad \cdot\chi^{r_5'\cdot H(\mathfrak{M}\|\mathfrak{E})}\cdot\kappa^{r_5'}, \\
\sigma_1 &= e(\vartheta^{H(ID)}\cdot\psi,\sigma_1') = e(\vartheta^{H(ID)}\cdot\psi,g^{r_2'}), \\
\sigma_2 &= g^{r_3}, \\
\sigma_3 &= g^{r_4}, \\
\sigma_4 &= g^{r_5'}.
\end{aligned}$$

Thus, $\Phi' = \{\sigma_0,\sigma_1,\sigma_2,\sigma_3,\sigma_4\}$ is a valid signature, where we assure that $\ell\cdot H(ID) \neq 0$ **mod** q and $\alpha\cdot H(\mathfrak{M}\parallel\mathfrak{E}) \neq 0$ **mod** q.

Forgery: If the algorithm $\mathcal{B}$ does not abort as a consequence of one of the queries above, the adversary $\mathcal{A}$ will, with probability at least ε, return a message $\mathfrak{M}^*$, an event identifier $\mathfrak{E}^*$ and a valid forgery, $\Phi^* = \{\sigma_0^*,\sigma_1^*,\sigma_2^*,\sigma_3^*,\sigma_4^*\}$ for RL_ID^* and RL_PK^*, where RL_ID^* is an identity list including all identities of the ring members belong to this ring and RL_PK^* is a public key list including all public keys of the ring members belong to this ring, where

$$\begin{aligned}
\sigma_0^* &= g_2^{a}\cdot\vartheta^{r_2^*\cdot H(ID^*)}\cdot\psi^{r_2^*}\cdot\varpi^{r_3^*} \\
&\quad \cdot\mu^{r_4^*\cdot H(RL_ID^*\|RL_PK^*)}\cdot\tau^{r_4^*} \\
&\quad \cdot\chi^{r_5^*\cdot H(\mathfrak{M}^*\|\mathfrak{E}^*)}\cdot\kappa^{r_5^*}, \\
\sigma_1^* &= g^{r_2^*}, \\
\sigma_2^* &= g^{r_3^*}, \\
\sigma_3^* &= g^{r_4^*}, \\
\sigma_4^* &= g^{r_5^*}.
\end{aligned}$$

And $\mathcal{A}$ did not query ***Generate-Key*** on input $ID^* \in RL\_ID^*$ and did not query ***Sign*** on inputs RL_ID^*, RL_PK^*, $\mathfrak{M}^*$ and $\mathfrak{E}^*$.

If $\ell\cdot H(ID^*) \neq 0$ **mod** q or $\alpha\cdot H(\mathfrak{M}^*\parallel\mathfrak{E}^*) \neq 0$ **mod** q, then the algorithm $\mathcal{B}$ will abort.

If $\ell\cdot H(ID^*) = 0$ **mod** q and $\alpha\cdot H(\mathfrak{M}^*\parallel\mathfrak{E}^*) = 0$ **mod** q, then the algorithm $\mathcal{B}$ computes and outputs $\frac{\sigma_0^*}{Q} = g^{a\cdot b}$, which is the solution to the given CDH problem, where $Q = g^{r_2^*\cdot H(ID^*)}\cdot g^{r_2^*\cdot\partial}\cdot g^{r_3^*\cdot\eta}\cdot g^{r_4^*\cdot\nu\cdot H(RL\_ID^*\|RL\_PK^*)}\cdot g^{r_4^*\cdot\lambda}\cdot g^{r_5^*\cdot H(\mathfrak{M}^*\|\mathfrak{E}^*)}\cdot g^{r_5^*\cdot\pi}$.

Now, we analyze the probability of the algorithm $\mathcal{B}$ not aborting. For the simulation to complete without aborting, we require that all ***Generate-Key*** queries will have $\ell\cdot H(ID) \neq 0$ **mod** q, all ***Sign*** queries will have $\ell\cdot H(ID) \neq 0$ **mod** q and $\alpha\cdot H(\mathfrak{M}\parallel\mathfrak{E}) \neq 0$ **mod** q and that $\ell\cdot H(ID^*) = 0$ **mod** q and $\alpha\cdot H(\mathfrak{M}^*\parallel\mathfrak{E}^*) = 0$ **mod** q in forgery. If the algorithm $\mathcal{B}$ does not abort, then the following conditions must hold:

1) $\ell\cdot H(ID_i) \neq 0$ **mod** q in ***Generate-Key*** queries, with i=1, 2.....q_g;
2) $\ell\cdot H(ID_i) \neq 0$ **mod** q and $\alpha\cdot H(\mathfrak{M}_i\parallel\mathfrak{E}_i) \neq 0$ **mod** q in ***Sign*** queries, with i=1, 2.....q_s;
3) The algorithm $\mathcal{B}$ does not abort in forgery, namely $\ell\cdot H(ID^*) = 0$ **mod** q and $\alpha\cdot H(\mathfrak{M}^*\parallel\mathfrak{E}^*) = 0$ **mod** q.

To make the analysis simpler, we will define the events E_i, F_i, T_i, R^*, F^* as

E_i :$\ell\cdot H(ID_i) \neq 0$ **mod** q, with i=1, 2.....q_g;
F_i :$\ell\cdot H(ID_i) \neq 0$ **mod** q, with i=1, 2.....q_s;
T_i :$\alpha\cdot H(\mathfrak{M}_i\parallel\mathfrak{E}_i) \neq 0$ **mod** q, with i=1, 2.....q_s;
R^* :$\ell\cdot H(ID^*) = 0$ **mod** q;
F^* :$\alpha\cdot H(\mathfrak{M}^*\parallel\mathfrak{E}^*) = 0$ **mod** q.

Then the probability of $\mathcal{B}$ not aborting is

$$\Pr(not\_abort) = \Pr\left(\bigcap_{i=1}^{q_g} E_i \wedge \bigcap_{i=1}^{q_s}(F_i\wedge T_i)\wedge R^*\wedge F^*\right).$$

It is easy to see that the events $\bigcap_{i=1}^{q_g} E_i$, $\bigcap_{i=1}^{q_s} F_i$, $\bigcap_{i=1}^{q_s} T_i$, R^* and F^* are independent. Then we may compute

$$\begin{aligned}
\Pr(\bigcap_{i=1}^{q_g} E_i) &= 1-\Pr(\bigcup_{i=1}^{q_g} \neg E_i) = 1-q_g\cdot\frac{1^k}{1^k\cdot q}\\
&= 1-\frac{q_g}{q};\\
\Pr(\bigcap_{i=1}^{q_s} F_i) &= 1-\Pr(\bigcup_{i=1}^{q_s} \neg F_i) = 1-q_s\cdot\frac{1^k}{1^k\cdot q}\\
&= 1-\frac{q_s}{q};\\
\Pr(\bigcap_{i=1}^{q_s} T_i) &= 1-\Pr(\bigcup_{i=1}^{q_s} \neg T_i) = 1-q_s\cdot\frac{1^k}{1^k\cdot q}\\
&= 1-\frac{q_s}{q};\\
\Pr(R^*) &= \frac{1^k}{1^k\cdot q}=\frac{1}{q};\quad \Pr(F^*)=\frac{1^k}{1^k\cdot q}=\frac{1}{q}.
\end{aligned}$$

Thus,

$$\Pr(not\_abort)=\Pr\left(\bigcap_{i=1}^{q_g} E_i\wedge\bigcap_{i=1}^{q_s}(F_i\wedge T_i)\wedge R^*\wedge F^*\right)$$

$$=\Pr(\bigcap_{i=1}^{q_g} E_i)\cdot\Pr(\bigcap_{i=1}^{q_s} F_i)\cdot\Pr(\bigcap_{i=1}^{q_s} T_i)\cdot\Pr(R^*)\cdot\Pr(F^*)$$

$$=\left(1-\frac{q_g}{q}\right)\cdot\left(1-\frac{q_s}{q}\right)^2\cdot\frac{1}{q^2}.$$

We can get that $\varepsilon'=(1-\frac{q_g}{q})\cdot(1-\frac{q_s}{q})^2\cdot\frac{\varepsilon}{q^2}$.

If the simulation does not abort, the adversary $\mathcal{A}$ will create a valid forgery with probability at least ε. The algorithm $\mathcal{B}$ can then compute $g^{a\cdot b}$ from the forgery as shown above. The time complexity of the algorithm $\mathcal{B}$ is dominated by the time for the exponentiations and multiplications in the queries. We assume that the time for integer addition and integer multiplication and the time for hash computation can both be ignored, then the time complexity of the algorithm $\mathcal{B}$ is

$$\hbar'=\hbar+O(q_g\cdot(10\cdot C_{exp}+3\cdot C_{mul})+q_s\cdot(15\cdot C_{exp}+11\cdot C_{mul})).$$

Type II:

Let **TCRS** be a traceable certificateless ring signature scheme of Section 5. Additionally, let $\mathcal{A}$ be an ($\hbar$, ε, q_g, q_s)-adversary attacking **TCRS**. From the adversary $\mathcal{A}$, we construct an algorithm $\mathcal{B}$, for (g, g^a, g^b)$\in\mathbb{G}_1$, the algorithm $\mathcal{B}$ is able to use $\mathcal{A}$ to compute $g^{a\cdot b}$. Thus, we assume the algorithm $\mathcal{B}$ can solve the CDH with probability at least ε' and in time at most $\hbar'$, contradicting the ($\hbar'$, ε')-CDH assumption. Such a simulation may be created in the following way:

Setup: The KGC system inputs a security parameter 1^k. Additionally, let $\mathbb{G}_1$ and $\mathbb{G}_2$ be groups of prime order q and g be a generator of $\mathbb{G}_1$ and let $e:\mathbb{G}_1\times\mathbb{G}_1\to\mathbb{G}_2$ denote the bilinear map. The size of the group is determined by the security parameter and we set $\mathbb{A}\subseteq\mathbb{Z}_q$ as the universe of identities. One hash function, $H:\{0,1\}^*\to\mathbb{Z}_{1^k\cdot q}$ can be defined and used to generate any integer value in $\mathbb{Z}_{1^k\cdot q}$ (where 1^k represents the corresponding decimal number).

Then the system parameters are generated as follows. The algorithm chooses $y\in\mathbb{Z}_q$ and then sets $g_1=g^y$ and $g_2=g^b$ with $b\in\mathbb{Z}_q$ ($\mathcal{B}$ doesn't know b). Also, the algorithm chooses ℓ, ∂, ν, λ, η, α and $\pi\in\mathbb{Z}_q$ and then sets $\vartheta=g_2^{\ell}\cdot g$, $\psi=g^{\partial}$, $\mu=g^{\nu}$, $\tau=g^{\lambda}$, $\chi=g_2^{\alpha}\cdot g$, $\kappa=g^{\pi}$ and $\varpi=g^{\eta}$. Finally, the system outputs the public parameters $TCRK$=($\mathbb{G}_1$, $\mathbb{G}_2$, e, g, g_1, g_2, ϑ, ψ, μ, τ, χ, κ, ϖ) and the master private key $spk=g_2^y$ to $\mathcal{A}$.

Additionally, the user u^* is a challenger, whose identity and public key respectively are ID^* and $pk_{\{ID^*\}}$, we set that the member secret of the user u^*, $sx_{\{ID^*\}}=\vartheta^{a\cdot H(ID^*)}\cdot\psi^a$ and that the public key of u^*, $pk_{\{ID^*\}}=g^a$ ($\mathcal{B}$ doesn't know a).

Queries: When running the adversary $\mathcal{A}$, the relevant queries can occur. The algorithm $\mathcal{B}$ answers these in the following way:

- Generate-Key Queries: Given the public parameters $TCRK$ and the identity ID of the ring member ($ID\in RL\_ID$ where RL_ID is an identity list), the algorithm $\mathcal{B}$ can construct a private key of the ring member u by the following computation (ID is the identity of u):
  - Set-Secret: The algorithm randomly chooses $r_0\in\mathbb{Z}_q$, computes the member secret $sx_{\{ID\}}=\vartheta^{r_0\cdot H(ID)}\cdot\psi^{r_0}$, where we assure that $H(ID)\neq 0$ **mod** q.
  - Generate-Public Key: The algorithm outputs the public key $pk_{\{ID\}}=g^{r_0}$, which is added to the public key ring RL_PK, where RL_PK is a public key list including all public keys of the ring members belong to this ring.
- Sign Queries: Given the public parameters $TCRK$, the identity list RL_ID ($ID\in RL\_ID$ where ID is the identity of the ring member that belongs to this ring), the public key list RL_PK, the message $\mathfrak{M}$ and the event identifier $\mathfrak{E}$, the algorithm $\mathcal{B}$ chooses random $r_2,r_3,r_4,r_5\in\mathbb{Z}_q$ and computes

$$\begin{aligned}
\sigma_0 &= g_2^y\cdot\vartheta^{r_2\cdot H(ID)}\cdot\psi^{r_2}\cdot\varpi^{r_3}\\
&\quad\cdot\mu^{r_4\cdot H(RL_ID\|RL_PK)}\cdot\tau^{r_4}\\
&\quad\cdot\chi^{r_5\cdot H(\mathfrak{M}\|\mathfrak{E})}\cdot\kappa^{r_5},\\
\sigma_1' &= g^{r_2},\\
\sigma_2 &= g^{r_3},\\
\sigma_3 &= g^{r_4},\\
\sigma_4 &= g^{r_5}.
\end{aligned}$$

Finally, the algorithm outputs a forgery $\Phi = \{\sigma_0, \sigma_1', \sigma_2, \sigma_3, \sigma_4\}$ to the adversary $\mathcal{A}$, where we maximize the adversary's advantage, σ_1' is passed to $\mathcal{A}$. Thus, $\Phi' = \{\sigma_0, \sigma_1 = e(\vartheta^{H(ID)} \cdot \psi, \sigma_1'), \sigma_2, \sigma_3, \sigma_4\}$ is a valid signature, where we assure that $H(ID) \neq 0$ **mod** q and $H(\mathfrak{M} \parallel \mathfrak{E}) \neq 0$ **mod** q.

Forgery: If the algorithm $\mathcal{B}$ does not abort as a consequence of one of the queries above, the adversary $\mathcal{A}$ will, with probability at least ε, return its forgeries, ($\mathfrak{M}_1^*$, $\mathfrak{E}^*$, Φ_1^*, RL_ID^*, RL_PK^*) and ($\mathfrak{M}_2^*$, $\mathfrak{E}^*$, Φ_2^*, RL_ID^*, RL_PK^*) for the challenger u^*, with $ID^* \in RL\_ID^*$, $pk_{\{ID^*\}} \in RL\_PK^*$, $\Phi_1^* = \{\sigma_{10}^*, \sigma_{11}^*, \sigma_{12}^*, \sigma_{13}^*, \sigma_{14}^*, \sigma_{15}^*, \sigma_{16}^*\}$ and $\Phi_2^* = \{\sigma_{20}^*, \sigma_{21}^*, \sigma_{22}^*, \sigma_{23}^*, \sigma_{24}^*, , \sigma_{25}^*, \sigma_{26}^*\}$, where

$$\begin{aligned}
\sigma_{10}^* &= g_2^y \cdot \vartheta^{(r_{12}^*+a)\cdot H(ID^*)} \cdot \psi^{r_{12}^*+a} \cdot \varpi^{r_{13}^*} \\
&\quad \cdot \mu^{r_{14}^* \cdot H(RL_ID^* \| RL_PK^*)} \cdot \tau^{r_{14}^*} \\
&\quad \cdot \chi^{r_{15}^* \cdot H(\mathfrak{M}_1^* \| \mathfrak{E}^*)} \cdot \kappa^{r_{15}^*}, \\
\sigma_{11}^* &= g^{r_{12}^*}, \\
\sigma_{12}^* &= g^{r_{13}^*}, \\
\sigma_{13}^* &= g^{r_{14}^*}, \\
\sigma_{14}^* &= g^{r_{15}^*}, \\
\sigma_{15}^* &= g_2^{r_{12}^*}, \\
\sigma_{16}^* &= g_2^{r_{15}^*}.
\end{aligned}$$

$$\begin{aligned}
\sigma_{20}^* &= g_2^y \cdot \vartheta^{(r_{22}^*+a)\cdot H(ID^*)} \cdot \psi^{r_{22}^*+a} \cdot \varpi^{r_{23}^*} \\
&\quad \cdot \mu^{r_{24}^* \cdot H(RL_ID^* \| RL_PK^*)} \cdot \tau^{r_{24}^*} \\
&\quad \cdot \chi^{r_{25}^* \cdot H(\mathfrak{M}_2^* \| \mathfrak{E}^*)} \cdot \kappa^{r_{25}^*}, \\
\sigma_{21}^* &= g^{r_{22}^*}, \\
\sigma_{22}^* &= g^{r_{23}^*}, \\
\sigma_{23}^* &= g^{r_{24}^*}, \\
\sigma_{24}^* &= g^{r_{25}^*}, \\
\sigma_{25}^* &= g_2^{r_{22}^*}, \\
\sigma_{26}^* &= g_2^{r_{25}^*}.
\end{aligned}$$

Remark 4. *In fact, σ_{11}^* should be equal to $g^{r_{12}^*} \cdot pk_{\{ID^*\}}$, σ_{21}^* should be equal to $g^{r_{22}^*} \cdot pk_{\{ID^*\}}$. Additionally, because the adversary $\mathcal{A}$ can compute $\sigma_{11}^* = g^{r_{12}^*}$ and $\sigma_{14}^* = g^{r_{15}^*}$, $\mathcal{A}$ can easily convert these computations to $\sigma_{15}^* = g_2^{r_{12}^*}$ and $\sigma_{16}^* = g_2^{r_{15}^*}$, where σ_{15}^* and σ_{16}^* return to the algorithm $\mathcal{B}$ so as to make $\mathcal{B}$ solve the CDH problem. Similarly, σ_{25}^* and σ_{26}^* also return to the algorithm $\mathcal{B}$.*

And the forgeries satisfy the following conditions:

1) $1 \leftarrow$ ***Verify***$(TCRK, RL\_ID^*, RL\_PK^*, \mathfrak{M}_1^*, \mathfrak{E}^*, \Phi_1^*)$;
2) $1 \leftarrow$ ***Verify***$(TCRK, RL\_ID^*, RL\_PK^*, \mathfrak{M}_2^*, \mathfrak{E}^*, \Phi_2^*)$;
3) $ID^* \leftarrow$ ***Trace-User***$(TCRK, RL\_ID^*, RL\_PK^*, \{\mathfrak{M}_1^*, \Phi_1^*\}, \{\mathfrak{M}_2^*, \Phi_2^*\}, \mathfrak{E}^*)$;
4) $\mathcal{A}$ did not query ***Generate-Key*** on input $ID^* \in RL\_ID^*$ and did not query ***Sign*** on inputs RL_ID^*, RL_PK^*, $\mathfrak{M}_1^*$ ($\mathfrak{M}_2^*$) and $\mathfrak{E}^*$.

So, the algorithm $\mathcal{B}$ computes and outputs

$$\frac{\sigma_{10}^*}{Q} = g_2^{\ell \cdot a \cdot H(ID^*)},$$

where $Q = g_2^y \cdot g_2^{r_{12}^* \cdot \ell \cdot H(ID^*)} \cdot g^{r_{12}^* \cdot H(ID^*)} \cdot pk_{\{ID^*\}}^{H(ID^*)} \cdot g^{r_{12}^* \cdot \partial} \cdot pk_{\{ID^*\}}^{\partial} \cdot g^{r_{13}^* \cdot \eta} \cdot g^{r_{14}^* \cdot \nu \cdot H(RL\_ID^* \| RL\_PK^*)} \cdot g^{r_{14}^* \cdot \lambda} \cdot g_2^{r_{15}^* \cdot \alpha \cdot H(\mathfrak{M}_1^* \| \mathfrak{E}^*)} \cdot g^{r_{15}^* \cdot H(\mathfrak{M}_1^* \| \mathfrak{E}^*)} \cdot g^{r_{15}^* \cdot \pi}$. Further, we can compute $\sqrt[\ell \cdot H(ID^*)]{g_2^{\ell \cdot a \cdot H(ID^*)}} = g_2^a = g^{a \cdot b}$, which is the solution to the given CDH problem.

Now, we analyze the probability of the algorithm $\mathcal{B}$ not aborting. For the simulation to complete without aborting, we require that all ***Generate-Key*** queries will have $H(ID) \neq 0$ **mod** q, all ***Sign*** queries will have $H(ID) \neq 0$ **mod** q and $H(\mathfrak{M} \parallel \mathfrak{E}) \neq 0$ **mod** q. If the algorithm $\mathcal{B}$ does not abort, then the following conditions must hold:

1) $H(ID_i) \neq 0$ **mod** q in ***Generate-Key*** queries, with i=1, 2.....q_g;
2) $H(ID_i) \neq 0$ **mod** q and $H(\mathfrak{M}_i \parallel \mathfrak{E}_i) \neq 0$ **mod** q in ***Sign*** queries, with i=1, 2.....q_s;

To make the analysis simpler, we will define the events E_i, F_i, T_i as

E_i :$H(ID_i) \neq 0$ **mod** q, with i=1, 2.....q_g;
F_i :$H(ID_i) \neq 0$ **mod** q, with i=1, 2.....q_s;
T_i :$H(\mathfrak{M}_i \parallel \mathfrak{E}_i) \neq 0$ **mod** q, with i=1, 2.....q_s;

Then the probability of $\mathcal{B}$ not aborting is

$$\Pr(not\_abort) = \Pr\left(\bigcap_{i=1}^{q_g} E_i \wedge \bigcap_{i=1}^{q_s} (F_i \wedge T_i)\right).$$

It is easy to see that the events $\bigcap_{i=1}^{q_g} E_i$, $\bigcap_{i=1}^{q_s} F_i$, $\bigcap_{i=1}^{q_s} T_i$ are independent. Then we may compute

$$\Pr(\bigcap_{i=1}^{q_g} E_i) = 1 - \Pr(\bigcup_{i=1}^{q_g} \neg E_i) = 1 - q_g \cdot \frac{1^k}{1^k \cdot q} = 1 - \frac{q_g}{q};$$

$$\Pr(\bigcap_{i=1}^{q_s} F_i) = 1 - \Pr(\bigcup_{i=1}^{q_s} \neg F_i) = 1 - q_s \cdot \frac{1^k}{1^k \cdot q} = 1 - \frac{q_s}{q};$$

$$\Pr(\bigcap_{i=1}^{q_s} T_i) = 1 - \Pr(\bigcup_{i=1}^{q_s} \neg T_i) = 1 - q_s \cdot \frac{1^k}{1^k \cdot q} = 1 - \frac{q_s}{q};$$

Thus,

$$\begin{aligned}
\Pr(not_abort) &= \Pr\left(\bigcap_{i=1}^{q_g} E_i \wedge \bigcap_{i=1}^{q_s} (F_i \wedge T_i)\right) \\
&= \Pr(\bigcap_{i=1}^{q_g} E_i) \cdot \Pr(\bigcap_{i=1}^{q_s} F_i) \cdot \Pr(\bigcap_{i=1}^{q_s} T_i)
\end{aligned}$$

$$= \left(1 - \frac{q_g}{q}\right) \cdot \left(1 - \frac{q_s}{q}\right)^2$$

We can get that $\varepsilon' = (1 - \frac{q_g}{q}) \cdot (1 - \frac{q_s}{q})^2 \cdot \varepsilon$.

If the simulation does not abort, the adversary $\mathcal{A}$ will create a valid forgery with probability at least ε. The algorithm $\mathcal{B}$ can then compute $g^{a \cdot b}$ from the forgery as shown above. The time complexity of the algorithm $\mathcal{B}$ is dominated by the time for the exponentiations and multiplications in the queries. We assume that the time for integer addition and integer multiplication and the time for hash computation can both be ignored, then the time complexity of the algorithm $\mathcal{B}$ is

$$\hbar' = \hbar + O(q_g \cdot (3 \cdot C_{exp} + C_{mul}) + q_s \cdot (12 \cdot C_{exp} + 7 \cdot C_{mul})).$$

Then, from the above proofs, we may get that

$$\varepsilon' = [(1 - \tfrac{q_g}{q}) \cdot (1 - \tfrac{q_s}{q})^2 \cdot \tfrac{\varepsilon}{q^2}] \parallel [(1 - \tfrac{q_g}{q}) \cdot (1 - \tfrac{q_s}{q})^2 \cdot \varepsilon],$$

$\hbar' = \max\{\hbar + O(q_g \cdot (10 \cdot C_{exp} + 3 \cdot C_{mul}) + q_s \cdot (15 \cdot C_{exp} + 11 \cdot C_{mul})), \hbar + O(q_g \cdot (3 \cdot C_{exp} + C_{mul}) + q_s \cdot (12 \cdot C_{exp} + 7 \cdot C_{mul}))\}$. Thus, Theorem 1 follows.

Theorem 2. *The scheme of Section 5 is a linkable (traceable) TCRS scheme when it satisfies the following condition—the scheme of Section 5 is ($\hbar$, ε, q_g, q_s)-secure, assuming that the ($\hbar'$, ε')-CDH assumption holds in $\mathbb{G}_1$, where*

$$\varepsilon' = [\left(1 - \tfrac{q_g}{q}\right) \cdot \left(1 - \tfrac{q_s}{q}\right)^2 \cdot \left(\prod_{i=0}^{i=t} \tfrac{1^k - i}{1^k \cdot q - i}\right)^2 \cdot \varepsilon] \parallel [(1 - \tfrac{q_g}{q}) \cdot (1 - \tfrac{q_s}{q})^2 \cdot \varepsilon],$$

$\hbar' = \max\{\hbar + O(q_g \cdot (10 \cdot C_{exp} + 3 \cdot C_{mul}) + q_s \cdot (15 \cdot C_{exp} + 11 \cdot C_{mul})), \hbar + O(q_g \cdot (3 \cdot C_{exp} + C_{mul}) + q_s \cdot (12 \cdot C_{exp} + 7 \cdot C_{mul}))\}$,
and q_g is the maximal number of "Generate-Key" oracle queries, q_s is the maximal number of "Sign" oracle queries, t is the number of user (ring member) private keys possessed by adversary, C_{mul} and C_{exp} are respectively the time for a multiplication and an exponentiation in $\mathbb{G}_1$.

Theorem 3. *The scheme of Section 5 is exculpable when it satisfies the following condition—the scheme of Section 5 is ($\hbar$, ε, q_g, q_s)-secure, assuming that the ($\hbar'$, ε')-CDH assumption holds in $\mathbb{G}_1$, where*

$$\varepsilon' = [(1 - \tfrac{q_g}{q}) \cdot (1 - \tfrac{q_s}{q})^2 \cdot \tfrac{\varepsilon}{q^2}] \parallel [(1 - \tfrac{q_g}{q}) \cdot (1 - \tfrac{q_s}{q})^2 \cdot \varepsilon],$$

$\hbar' = \max\{\hbar + O(q_g \cdot (10 \cdot C_{exp} + 3 \cdot C_{mul}) + q_s \cdot (15 \cdot C_{exp} + 11 \cdot C_{mul})), \hbar + O(q_g \cdot (3 \cdot C_{exp} + C_{mul}) + q_s \cdot (12 \cdot C_{exp} + 7 \cdot C_{mul}))\}$,
and q_g is the maximal number of "Generate-Key" oracle queries, q_s is the maximal number of "Sign" oracle queries, C_{mul} and C_{exp} are respectively the time for a multiplication and an exponentiation in $\mathbb{G}_1$.

Theorem 4. *The scheme of Section 5 is ($\hbar$, ε, q_g, q_s)-anonymous, assuming that the ($\hbar'$, ε')-CDH assumption holds in $\mathbb{G}_1$, where*

$$\varepsilon' = [(1 - \tfrac{q_{g_1}}{q}) \cdot (1 - \tfrac{q_{s_1}}{q})^2 \cdot (1 - \tfrac{q_{g_2}}{q}) \cdot (1 - \tfrac{q_{s_2}}{q})^2 \cdot \tfrac{\varepsilon}{q^2}] \parallel [(1 - \tfrac{q_{g_1}}{q}) \cdot (1 - \tfrac{q_{s_1}}{q})^2 \cdot (1 - \tfrac{q_{g_2}}{q}) \cdot (1 - \tfrac{q_{s_2}}{q})^2 \cdot \varepsilon],$$

$\hbar' = \max\{\hbar + O((q_{g_1} + q_{g_2}) \cdot (7 \cdot C_{exp} + C_{mul}) + (q_{s_1} + q_{s_2}) \cdot (15 \cdot C_{exp} + 11 \cdot C_{mul})), \hbar + O((q_{g_1} + q_{g_2}) \cdot (3 \cdot C_{exp} + C_{mul}) + (q_{s_1} + q_{s_2}) \cdot (12 \cdot C_{exp} + 7 \cdot C_{mul}))\}$,
q_{g_1} and q_{g_2} are respectively the maximal numbers of "Generate-Key" oracle queries in the Queries Phases 1 and 2, q_{s_1} and q_{s_2} are respectively the maximal numbers of "Sign" oracle queries in the Queries Phases 1 and 2, C_{mul} and C_{exp} are respectively the time for a multiplication and an exponentiation in $\mathbb{G}_1$.

7 Conclusions

In this paper, we present a fully traceable certificateless ring signature scheme, which has a security reduction to the computational Diffie-Hellman assumption. Also, we give a formal security model for traceable certificateless ring signature. Under our security model, the proposed scheme is proved to have the properties of anonymity and traceability with enough security. Compared with other traceable ring signature schemes, the proposed scheme is efficient. However, because the proposed scheme is not enough efficient in computing linking of signatures, the work about TCRS still needs to be further progressed.

Acknowledgments

This study is funded by the National Natural Science Foundations of China (No.61402055, No.61504013), the Hunan Natural Science Foundation (2018JJ2445) and the Hunan Provincial Scientific Research Project (No.15C0041,No.12C0010). The authors gratefully acknowledge the anonymous reviewers for their valuable comments.

References

[1] M. Abe, M. Ohkubo and K. Suzuki, "1-out-of-n signatures from a variety of keys," in *International Conference on the Theory and Application of Cryptology and Information Security*, pp. 415–432, 2002.

[2] M. Abe, M. Ohkubo and K. Suzuki, "Efficient threshold signer-ambiguous signatures from variety of keys," *IEICE Transactions on Fundamentals of Electronics Communications and Computer Sciences*, vol. E87-A, no. 2:471–479, 2004.

[3] S. S. Al-Riyami, K. G. Paterson, "Certificateless public key cryptography," in *International Conference on the Theory and Application of Cryptology and Information Security*, pp. 452–473, 2003.

[4] M. H. Au, S. S. M. Chow, W. Susilo and P. P. Tsang, "Short linkable ring signatures revisited," in *European Public Key Infrastructure Workshop*, pp. 101–115, 2006.

[5] M. H. Au, J. K. Liu, W. Susilo, T. H. Yuen, "Secure ID-based linkable and revocable-iff-linked ring signature with constant-size construction," *Preprint Submitted to Theoretical Computer Science*, pp. 1-14, vol. 469, Apr. 23, 2013.

[6] M. H. Au, J. K. Liu, W. Susilo and T. H. Yuen, "Constantsize ID-based linkable and revocable-iff-linked ring signature," in *International Conference on Cryptology in India*, pp. 364–378, 2006.

[7] M. H. Au, J. K. Liu, T. H. Yuen, D. S. Wong, "ID-based ring signature scheme secure in the standard model", in *Proceeding of IWSEC 2006*, pp.1–16, 2006.

[8] A. K. Awasthi, S. Lal, "ID-based ring signature and proxy ring signature schemes from bilinear pairings," *International Journal of Network Security*, vol.4, no.2, pp. 187-192, Mar. 2007.

[9] P. S. L. M. Barreto, B. Libert, N. McCullagh, J. Quisquater, "Efficient and provably-secure identity-based signatures and signcryption from bilinear maps," in *International Conference on the Theory and Application of Cryptology and Information Security*, pp. 515–532, 2005.

[10] A. Bender, J. Katz and R. Morselli, "Ring signatures:stronger definitions and constructions without random oracles," in *Theory of Cryptography Conference*, pp. 60–79, 2006.

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

[12] D. Boneh, C. Gentry, B. Lynn and H. Shacham, "Aggregate and verifieably encrypted signatures from bilinear maps," in *International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 416–432, 2003.

[13] D. Boneh, M. Hanburg, "Generalized identity based and broadcast encryption schemes," in *International Conference on the Theory and Application of Cryptology and Information Security*, pp. 455–470, 2008.

[14] S. Brands, "Untraceable off-line cash in wallet with observers," in *Annual International Cryptology Conference*, pp. 302–318, 1993.

[15] E. Bresson, J. Stern and M. Szydlo, "Threshold ring signatures and applications to Ad-hoc groups," in *Annual International Cryptology Conference*, pp. 465–480, 2002.

[16] J. C. Cha, J. H. Cheon, "An identity-based signature from gap diffie-hellman groups," in *International Workshop on Public Key Cryptography*, pp. 18–30, 2002.

[17] C. C. Chang, C. Y. Sun, S. C. Chang, "A strong RSA-based and certificateless-based signature scheme," *International Journal of Network Security*, vol.18, no.2, pp.201-208, Mar. 2016.

[18] D. Chaum, "Blind signatures for untraceable payments," in *Advances in Cryptology*, pp. 199–204.

[19] D. Chaum, A. Fiat and M. Naor, "Untraceable electronic cash," in *Conference on the Theory and Application of Cryptography*, pp. 319–327, 1990.

[20] D. Chaum and E. Van Heyst, "Group signatures," in *Workshop on the Theory and Application of of Cryptographic Techniques*, pp. 257–265, 1991.

[21] S. S. M. Chow, J. K. Liu and D. S. Wong, "Robust receipt-free election system with ballot secrecy and verifieability," in *Network and Distributed System Security Symposiumndss*, 2008.

[22] S. S. M. Chow, S. M. Yiu and L. C. K. Hui, "Efficient Identity Based Ring Signature," in *Applied Cryptography and Network Security*, pp. 499–512, 2005.

[23] I. Damgard, K. Dupont and M. Pedersen, "Unclonable group identification," in *Advances in Cryptology*, pp. 555–572, 2006.

[24] Y. Dodis, A. Kiayias, A. Nicolosi and V. Shoup, "Anonymous identification in Ad hoc groups," in *Advances in Cryptology*, pp. 609–626, 2004.

[25] Y. Dodis, A. Kiayias, A. Nicolosi and V. Shoup, "Anonymous identification in Ad hoc groups," in *Advances in Cryptology*, pp. 609–626, 2004.

[26] K. Emura, A. Miyaji, K. Omote, "An r-Hiding revocable group signature scheme: Group signatures with the property of hiding the number of revoked users," *Journal of Applied Mathematics*, vol. 2014, pp. 14, 2014.

[27] E. Fujisaki, K. Suzuki, "Traceable ring signature," in *International Workshop on Public Key Cryptography*, pp. 181–200, 2007.

[28] E. Fujisaki, "Sub-linear size traceable ring signatures without random oracles," in *Cryptographers Track at the RSA Conference*, pp. 393–415, 2011.

[29] D. He, M. K. Khan, S. Wu, "On the Security of a RSA-based Certificateless Signature Scheme," *International Journal of Network Security*, vol. 16, no. 1, pp. 78-80, Jan. 2014.

[30] F. Hess, "Efficient identity based signature schemes based on pairings," in *International Workshop on Selected Areas in Cryptography*, pp. 310–324, 2003

[31] L. Ibraimi, S. Nikova, P. Hartel, W. Jonker, "An Identity-Based Group Signature with Membership Revocation in the Standard Model," *Faculty of Electrical Engineering, Mathematics & Computer Science*, pp. 16, 2010.

[32] I. R. Jeong, J. O. Kwon, D.g H. Lee, "Analysis of revocable-iff-linked ring signature scheme," *IEICE Transactions on Fundamentals of Electronics Communications & Computer Sciences*, pp.322–325, 2009.

[33] Y. Komano, K. Ohta, A. Shimbo and S. Kawamura, "Toward the fair anonymous signatures: Deniable ring signatures," in *Cryptographers Track at the RSA Conference*, pp. 174–191, 2006.

[34] F. Laguillaumie and D. Vergnaud, "Multi-designated Verifiers Signatures," in *Information and Communications Security*, pp. 495–507, 2004.

[35] J. K. Liu, V. K. Wei and D. S. Wong, "Linkable spontaneous anonymous group signature for ad hoc groups (extended abstract)," in *Information Security and Privacy*, pp. 325–335, 2004.

[36] J. K. Liu and D. S. Wong, "Linkable ring signatures: Security models and new schemes," in *International Conference on Computational Science and Its Applications*, pp. 614–623, 2005.

[37] J. K. Liu and D. S. Wong, "Enhanced security models and a generic construction approach for linkable ring signature," *International Journal of Foundations of Computer Science*, vol. 17, no. 6, pp. 1403–1422, 2006.

[38] E. J. L. Lu, M. S. Hwang, and C. J. Huang, "A new proxy signature scheme with revocation", *Applied Mathematics and Computation*, vol. 161, no. 3, PP. 799-806, Feb. 2005.

[39] M. Naor, "Deniable ring authentication," in *Annual International Cryptology Conference*, pp. 481–498, 2002.

[40] T. Okamoto and K. Ohta, "Universal electronic cash," in *Annual International Cryptology Conference*, pp. 324-337, 1992.

[41] K. G. Paterson, J. C. N. Schuldt, "Efficient identity-based signatures secure in the standard model," in *Information Security and Privacy*, pp.207–222, 2006.

[42] R. Rivest, A. Shamir and Y. Tauman, "How to leak a secret," in *Advances in Cryptology*, pp. 552–565, 2001.

[43] H. Singh, G. K. Verma, "ID-based proxy signature scheme with message recovery," *Journal of Systems and Software*,pp. 209–214, 2012.

[44] W. Susilo and Y. Mu, "Non-interactive deniable ring authentication," in *Information Security and Cryptology*, pp. 386–401, 2004.

[45] P. P. Tsang and V. K. Wei, "Short linkable ring signatures for e-voting, e-cash and attestation," in *Information Security Practice and Experience*, pp. 48–60, 2005.

[46] P. P. Tsang, V. K. Wei, T. K. Chan, M. H. Au, J. K. Liu and D. S. Wong, "Separable linkable threshold ring signatures," in *Progress in Cryptology*, pp. 389–398, 2004.

[47] B. Waters, "Efficient identity-based encryption without random oracles," in *Advances in Cryptology*, pp.114–127, 2005.

[48] F. T. Wen, S.J. Cui, J. N. Cui, "An ID-based proxy signature scheme secure against proxy key exposure," *International Journal of Advancements in Computing Technology*, pp. 108–116, 2011.

[49] D. S. Wong, K. Fung, J. K. Liu and Victor K. Wei, "On the RS-code construction of ring signature schemes and a threshold setting of RST," in *Information and Communications Security*, pp. 34–46, 2003.

[50] W. Wu, Y. Mu, W. Susilo, J. Seberry, X.Y. Huang, "Identity-based proxy signature from pairings," in *International Conference on Autonomic and Trusted Computing*, pp. 22–31, 2007.

[51] F. Zhang and K. Kim, "ID-Based blind signature and ring signature from pairings," in *International Conference on the Theory and Application of Cryptology and Information Security*, pp. 533–547, 2002.

[52] D. Zheng, X. Li, K. Chen and J. Li, "Linkable ring signatures from linear feedback shift register," in *International Conference on Embedded and Ubiquitous Computing*, pp. 716–727, 2007.

[53] D. Y. W. Liu, J. K. Liu, Y. Mu, W. Susilo and D. S. Wong, "Revocable ring signature," *Journal of Computer Science and Technology*, vol. 22, no. 6, pp. 785–794, 2007.

Biography

Ke Gu received his Ph.D. degree in School of Information Science and Engineering from Central South University in 2012. He is currently a Lecturer at Changsha University of Science and Technology. His research interests include cryptography, network and information security.

LinYu Wang is pursuing her master degree. Her research interests include social network, network and information security.

Na Wu is pursuing her master degree. Her research interests include fog computing, network and information security.

NianDong Liao is currently a Lecturer at Changsha University of Science and Technology. His research interests include cloud computing, network and information security.

An Untraceable Voting Scheme Based on Pairs of Signatures

Kazi Md. Rokibul Alam[1], Adnan Maruf[1], Md. Rezaur Rahman Rakib[1], G. G. Md. Nawaz Ali[1,2], Peter Han Joo Chong[3], and Yasuhiko Morimoto[4]

(Corresponding author: G. G. Md. Nawaz Ali)

Department of CSE, Khulna University of Engineering & Technology, Khulna 9203, Bangladesh[1]
School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore[2]
Department of Electrical and Electronic Engineering, Auckland University of Technology, New Zealand[3]
Graduate School of Engineering, Hiroshima University, Higashi-Hiroshima 739-8521, Japan[4]
(Email: nawaz.ali@ntu.edu.sg)

(Received May 10, 2017; revised and accepted Sept. 9, 2017)

Abstract

This paper proposes a new electronic voting (e-voting) scheme that exploits 2 pairs of signatures of signing (election) authorities. One pair of signatures on each voter's same blinded token enables the voter to appear to authorities in consecutive election stages anonymously. Another pair of signatures on each voter's same blinded vote enables authorities to protect them from the voter's dishonesties. Namely, while a vote remains same within its 2 different signed forms, the voter cannot claim that her vote is disrupted by other entities while intentionally submitting a meaningless or invalid vote. The scheme is suitable for small community where the number of voters is not very high. Here for vote construction, Hwang *et al.*'s untraceable blind signature (BS) scheme is exploited. Thereby no mutually independent signing authority involved in the scheme is able to link the resulting vote-signature pair even when the signature is publicly revealed. When compared with existing schemes, the proposed scheme requires straightforward computations and minimal assumptions regarding trustworthiness, i.e., nothing can make the scheme unreliable while only a single authority is honest among multiple authorities. Moreover, it achieves major security aspects of e-voting in a simple way, namely, it conforms privacy, accuracy, un-reusability, fairness, universal verifiability, dispute-freeness, robustness, incoercibility and scalability

Keywords: Anonymous Credential; Electronic Voting; RSA; Signature Pairs; Untraceable Blind Signature

1 Introduction

Voting is the basic instrument to sustain democracy in any society. It authorizes an official mechanism for the people to express their views to the government. The conventional procedure of the voting system claims the voter to come in person to vote which results in low participation rate. 'Vote by e-mail' system has evolved for increasing the participation rate, especially, in the sparsely populated area. However, this process is time consuming for the authority because it demands extra effort for collecting and counting ballots manually [8]. With the promotion of computing devices, computer networks and cryptographic protocols; electronic voting (e-voting) scheme can be designed to overcome troubles of the conventional procedure. Moreover, election process can be made more appropriate and convenient by using e-voting scheme for the voter to vote at any time and place [19].

An ideal e-voting scheme must satisfy privacy, eligibility, un-reusability, accuracy, fairness, universal verifiability, dispute-freeness, receipt-freeness, robustness, incoercibility, practicality and scalability [14, 17, 24]. Among them, practicality and scalability are related with the implementation of the scheme, whereas others are regarded as security requirements. Without fulfilling these requirements, prevalent fraud and corruption may take place in the election. Nonetheless, attaining all of the requirements is a challenge. Moreover, compared to the traditional voting scheme, e-voting scheme is more vulnerable because it requires digital processing of election data.

This paper proposes a new e-voting scheme that employs 2 pairs of signatures of signing (election) authorities. A pair of signatures on each voter's same blinded vote is generated by multiple mutually independent signing authorities to ensure the correctness of vote construction and the honesty of authorities. Namely, even when unblinded signed vote in 2 different forms are meaningless, it ensures that the vote is meaningless from the beginning because it is impossible for an unauthorized entity to generate the signature pair of multiple authorities consistently. In addition, another pair of signatures on each

voter's same blinded token enables a voter to appear in consecutive election stages anonymously. Moreover, to enable a voter to be a registered one anonymously; the scheme adopts anonymous tag based credential proposed in [33].

The rest of this paper is organized as follows. Section 2 summarizes several related works with justification of the proposed scheme. Section 3 explains the cryptographic building blocks required to develop the proposed scheme. Section 4 states the configuration, Section 5 represents an overview and Section 6 illustrates the individual stages of the scheme. Section 7 discusses the performance analysis, and Section 8 describes the security analysis of the scheme. Finally, Section 9 concludes the paper.

2 Related Works

Extensive researches on e-voting schemes have been conducted till now. Recently, various homomorphic encryption, blind signature (BS) and mixnet based voting schemes have been proposed along with different cryptographic techniques. Several schemes achieve receipt-freeness by exploiting specialized hardware like tamper resistant randomizer (TRR) [20]. Moreover, to ensure the correctness of votes, they deploy zero knowledge proof (ZKP), which requires significant computations. Again, in these schemes, through specialized devices, authorities may figure out the random number of a voter and use it to link the voter which results that these schemes are not completely receipt-free. Although the criterion of TRR proposed in [20] is such that the voter who exploits it finally loses her knowledge on randomness, here TRR has impaired the practicality of this scheme. The scheme proposed in [3] satisfies major security requirements, and its deployed cryptosystem supports probabilistic, homomorphic and commutative [16] properties altogether. However, because of its exploited cryptosystem, keys of involved entities required for both encryption decryption and signing verification must be kept as secret. Therefore a voter needs to interact with authorities while encrypting her vote and/or confirming the correctness of encryption and signing operations. These increase the computation and communication overheads of involved entities, and make the scheme unscalable. The scheme proposed in [21] named as 'proxy e-voting scheme' exploits proxy signature to enable a voter to delegate a proxy to cast her vote. However because of its 'double voting detection' capability, while double voting occurs, the authority can identify the responsible voter. Thereby the link between the vote and its voter is revealed which sacrifices the privacy of the voter. Another scheme known as Helios [2], is the first web based, open auditing system that satisfies both individual and universal verifiability, but cannot provide a strong guarantee of privacy. It runs as a client program in a browser, and a voter can submit her vote by using the browser. Finally, while vote submission closes, it shuffles all encrypted votes to disable the link between a vote and its voter, and produces a non-interactive ZKP to prove the correctness of shuffling. In contrast, the vote construction procedure in our proposed scheme deploys public keys of signing authorities. Note that our scheme does not use either any complicated protocol like ZKP or any specialized hardware or software. Moreover, it ensures the privacy of the voter, does not reveal her identity in any circumstances, even if she submits a meaningless vote to disrupt voting.

E-voting schemes based on BS are simple and efficient to implement, support flexible vote formats and do not exploit complicated ZKP. But the voter's blinding factors can be used as a receipt of the vote and thereby the receipt-freeness is sacrificed. Also, since every vote is blinded and unblinded only by its corresponding voter, this yields universal verifiability [14, 28]. A scheme proposed in [11] is based on Chaum's BS. Herein while voting, a registered voter submits her unblinded signed vote anonymously. Later on, a list of received ballots is published that is accessible by all voters. Finally in order to decrypt the vote, each voter needs to interact with the tallying authority by sending her private key. Although the scheme satisfies privacy, fairness, scalability, etc; its' major limitation is that the registration authority can detect the abstaining registered voters and can add votes for them. The scheme proposed in [32] exploits a uniquely threshold BS to get blind threshold votes, and allows any registered voter to abstain from vote submission. It also uses threshold cryptosystem to guarantee the fairness among the candidates campaign. Although it satisfies practicality, scalability and robustness; it can achieve fairness and accuracy conditionally. Another scheme proposed in [6] deploys pseudo-voter identity (PVID) developed by Chaum's BS to ensure the voter's anonymity. It does not use other complex cryptographic algorithms like ZKP or homomorphic encryption, and has no physical assumptions such as untappable channels. However, it has shortcoming, i.e., while ballot generator, key generator and counter work together and conspire, they can modify casted votes. Also there is possibility that corrupted authority may trace the voter's IP over the internet. Moreover, the scheme is not so robust and can satisfy fairness and practicality conditionally. In contrast, though our proposed scheme is also based on BS, it deploys Hwang *et al.*'s BS which is utterly untraceable. Also, it engages multiple mutually independent signing authorities; thereby nothing can make the scheme unreliable while at least a single authority is honest. Moreover herein, since data about interactions among entities are publicly verifiable; disputes are resolvable.

Recently proposed some other schemes are Civitas [9], UVote [1], Cobra [4, 10] *etc.* Among them, Civitas [9] is based on the mechanism proposed in [18] and aims to satisfy both verifiability and incoercibility. However to attain incoercibility, it allows the voter to submit multiple votes where multiple votes with the same token are excluded during the tallying. Herein, each voter needs to include ZKPs indicating which earlier votes to be erased as well as

showing the knowledge of the choice and the token used in earlier votes. The scheme proposed in [4] also exploits ZKP. Although here incoercibility is achieved; unfortunately scalability and accuracy are traded-off. UVote [1] allows a registered voter to submit multiple votes from which only the last vote is counted, and thus satisfies incoercibility. Here initially a voter needs to register her primary account, and later on can add multiple accounts. But for verification, any notification and message is sent only to the primary account and it cannot be deleted online. Thus although verifiability is achieved, receipt-freeness is sacrificed because a receipt is provided to the voter. In Cobra [10], a registered voter's encrypted credential is attached with an encrypted bloom filter. The voter selects certain number of candidate passwords and registers anyone of them. Later on, the voter encrypts her vote using the registered password to regenerate the credential. Herein, as the voter can provide a fake or a panic password to the coercer and thereby he is unable to manipulate the voter; incoercibility is achieved but thereby verifiability is traded. On the contrary, our proposed scheme does not allow a voter either to use a fake credential or to submit her vote multiple times. Each voter appears to authorities for submitting and approving her vote anonymously. Also it exploits a pair of signatures of signing authorities on each voter's same blinded vote, i.e., each vote is constructed in 2 different forms that ensures the honesty of authorities.

There are some schemes known as paper based cryptographic voting schemes which are based on visual cryptography [5, 27]. However herein; a voter needs to envoy her computations in the voting booth. Therefore, the booth can easily identify the vote of a voter. Again, the paper ballots prepared in advance do not ensure privacy against its creators' [27]. Considering commercial e-voting scheme, Sandler *et al.* [30] have developed voting scheme which is based on cryptographic techniques and hardware/machines, like optical scan voting machine, direct/digital-recording electronic (DRE), *etc.* Being different, our proposed scheme is based on pairs of signatures, which does not require any complicated protocol, or any specialized hardware, but still it can provide a reliable voting scheme while only a single authority is honest among multiple authorities.

3 Cryptographic Building Blocks

The proposed scheme exploits several cryptographic tools. These are: Hwang *et al.*'s BS [25] for blinded signed vote construction, and Chaum' BS [7] for blinded signed token generation. Also, a pair of signatures on each voter's same blinded token is generated by signing authorities. Moreover, a pair of signatures of signing authorities on each voter's same blinded vote is generated. Besides while token acquisition, to authenticate a voter anonymously many mechanisms [13, 26, 33] are available and any one can be used, namely anonymous tag based credential proposed in [33]. This section describes the major cryptographic tools. Further, important notations that are used in this paper are summarized in Table 1.

3.1 Chaum's Blind Signature

Chaum's BS proposed in [7] is based on RSA cryptosystem and consists of five phases which are briefly described as follows.

1) *Initializing phase:* The signer (i.e., herein an election authority TM_i) randomly chooses 2 large primes p and q, and computes $n = p * q$ and $\varphi(n) = (p-1)*(q-1)$. The authority chooses 2 large numbers e and d such that $ed \equiv 1 \ mod \ \varphi(n)$ and the greatest common divisor (GCD) $(e, \varphi(n)) = 1$. Let (e, n) be the authority's public key and d be the authority's private signing key. The authority keeps (p, q, d) secure and publishes (e, n).

2) *Blinding phase:* The voter V_j has a message (i.e., herein the token T_j), and she wishes to have it signed by the authority. Now V_j randomly selects an integer r_j as the blinding factor, and computes the integer $\alpha = r_j^e * T_j \ mod \ n$ and submits it to the authority.

3) *Signing phase:* After receiving α from V_j, the authority computes the integer $t = \alpha^d \ mod \ n$ and sends it to V_j.

4) Unblinding phase: After receiving t from the authority, voter V_j computes $s = t * r_j^{-1} \ mod \ n$.

5) *Verifying phase:* As a result, s is the signature on the token T_j. Now anyone can verify the legitimacy of the signature by checking whether $s^e \equiv T_j \ mod \ n$.

Signature pairs on blinded token discussed in Section 3.3 is constructed based on this BS because cryptographic operations involved in its various phases are straightforward and their computations are also faster than that of Hwang *et al.*'s BS. Although it has some limitations [25], it is capable to conduct the registration stage (as discussed in Section 6.2) of the proposed scheme. Therefore instead of Hwang *et al.*'s BS, it is chosen here.

3.2 Hwang *et al.*'s Blind Signature

Hwang *et al.*'s BS proposed in [25] is also based on RSA cryptosystem. The advantage of this BS is that it satisfies requirements of an ideal BS scheme and specially overcomes the limitation of untraceability of Chaum's BS. Although a great number of BS schemes are available, most of them are unable to satisfy untraceability [25]. There are some untraceable BS schemes based on discrete logarithm problem proposed in [22,23]. However for vote construction, RSA based Hwang *et al.*'s BS is chosen for our proposed e-voting scheme. This is because, RSA based schemes are by far the easiest to understand and implement among all the public-key algorithms proposed over

Table 1: List of notations used in this paper

Notation	Description
V_j	Any Voter
v_j	Vote of V_j
T_j, r_j	Token and secret integer of V_j to blind T_j
ID_j, P/W_j	Identity and password of V_j
$T_j(A, ID_j, Z_j)$	Anonymous credential of V_j
$Z_j, U_j^{Z_j}$	A secret integer and used seal of V_j
A	Credential issuer
VM	Voting manger
TMs or $TM_1, \cdots, TM_P$	$P(\geq 2)$ Tallying managers
$e_{(1*)}, e_{(2*)}$	To blind T_j, 1st and 2nd form of public keys of $TM_1, \cdots, TM_P$
$d_{(1*)}, d_{(2*)}$	To sign on blinded T_j, 1st and 2nd form of signing keys of $TM_1, \cdots, TM_P$
$\alpha_{*1(rj,Tj)}$, $\alpha_{*2(rj,Tj)}$	1st and 2nd form of blinded T_j of V_j
$t(d_{(1*)}, \alpha_{*1}(r_j, T_j))$, $t(d_{(2*)}, \alpha_{*2}(r_j, T_j))$	1st and 2nd form of blinded signed T_j of V_j
$s(d_{(1*)}, T_j), s(d_{(2*)}, T_j)$	1st and 2nd form of unblinded signed T_j of V_j
$(r_{1j}, r_{2j}), (a_{1j}, a_{2j})$	Pair of secret integers and primes of V_j to blind v_j
$\{b_{(1*)}, b_{(2*)}\}$, $\{b'_{(1*)}, b'_{(2*)}\}$	2 pairs of primes of TM_1, $\cdots$, TM_P to sign on blinded v_j in 2 different forms
$e'_{(1*)}, e'_{(2*)}$	To blind v_j, 1st and 2nd form of public keys of $TM_1, \cdots, TM_P$
$d'_{(1*)}, d'_{(2*)}$	To sign on blinded v_j, 1st and 2nd form of signing keys of $TM_1, \cdots, TM_P$
$\{(w_{11j}, \cdots, w_{1Pj}), (u_{11j}, \cdots, u_{1Pj})\}$, $\{(w_{21j}, \cdots, w_{2Pj}), (u_{21j}, \cdots, u_{2Pj})\}$	2 pairs of integers of V_j to unblind v_j
$\alpha_{1*j}, \alpha_{2*j}$	1st and 2nd form of blinded v_j of V_j
$t_1(d'_{(1*)}, (\alpha_{1*j}, \alpha_{2*j}))$, $t_2(d'_{(2*)}, (\alpha_{1*j}, \alpha_{2*j}))$	1st and 2nd forms of blinded signed v_j of V_j
$s_1(d'_{(1*)}, v_{j*})$, $s_2(d'_{(2*)}, v_{j*})$	1st and 2nd form of unblinded signed v_j of V_j
BBs	Bulletin Boards

the years [31]. This BS also consists of five phases which are described as follows.

1) Initializing phase: This phase is same as the initializing phase in Chaum's BS. The authority TM_i keeps (p, q, d) secure where d is the authority's secret signing key and publishes (e, n) as public key.

2) Blinding phase: The voter V_j has a message (i.e., herein the vote v_j), and she wishes to have it signed by the authority. For this purpose, V_j randomly selects 2 distinct integers' r_1 and r_2 as the blinding factors. Then she randomly chooses 2 primes a_1 and a_2 such that $a_1 \neq a_2$ and $GCD\,(a_1, a_2)$, is 1. Then, V_j computes the blinded messages $\alpha_1 = r_1^e * v_j^{a_1}\ mod\ n$ and $\alpha_2 = r_2^e * v_j^{a_2}\ mod\ n$, and sends (α_1, α_2) to the authority.

3) Signing phase: After receiving (α_1, α_2) from V_j, the authority randomly chooses 2 primes b_1 and b_2 such that $b_1 \neq b_2$ and $GCD\,(b_1, b_2)$ is 1, and signs the blinded vote by computing $t_1 = \alpha_1^{(b_1 d)}\ mod\ n$ and $t_2 = \alpha_2^{(b_2 d)}\ mod\ n$. Then the authority sends them back to V_j along with (b_1, b_2). Note that (t_1, t_2, b_1, b_2) denote the blinded signatures.

4) Unblinding phase: After receiving (t_1, t_2, b_1, b_2) from the authority, voter V_j computes $a_1 b_1$ and $a_2 b_2$. Due to the four distinct primes (a_1, a_2, b_1, b_2) where $GCD\,(a_1, a_2) = 1$ and $GCD\,(b_1, b_2) = 1$, $GCD\,(a_1 b_1, a_2 b_2)$ is also equal to 1. When $GCD\,(a_1 b_1, a_2 b_2) = 1$, there must be exactly 2 integers w and u that satisfy the equation $a_1 b_1 w + a_2 b_2 u = 1$. It is called the Extended Euclidean algorithm. The four parameters (a_1, a_2, w, u) are kept secret by the V_j. Now the V_j computes $s_1 = t_1 * r_1^{-b_1} = v_j^{a_1 b_1 d}\ mod\ n$ and $s_2 = t_2 * r_2^{-b_2} = v_j^{a_2 b_2 d}\ mod\ n$. Then V_j can derive the signature s by computing $s = s_1^w * s_2^u\ mod\ n$ and publishes (v_j, s).

5) Verifying phase: As a result, s is the signature on the vote v_j. Now anyone can verify the legitimacy of the signature by checking whether $s^e \equiv v_j\ mod\ n$. In the following the notation $(mod\ n)$ is omitted.

Signature pairs on blinded vote discussed in Section 3.4 is constructed based on this scheme. As the scheme is completely untraceable, no one can know the link between the blinded signed vote of a voter and its corresponding unblinded signed form, therefore it is chosen here.

3.3 Signature Pairs on Blinded Token

Voter can act without disclosing her identity while showing her eligibility by using token. To prove her eligibility anonymously, voter V_j blinds her token T_j in 2 different sets i.e., calculates $\alpha_{*1}(r_j, T_j) = \{\alpha_{11}(r_j, T_j), \cdots, \alpha_{1P}(r_j, T_j)\} = \{(r_j^{e_{11}} * T_j), \cdots, (r_j^{e_{1P}} * T_j)\}$ and $\alpha_{*2}(r_j, T_j) = \{\alpha_{21}(r_j, T_j), \cdots, \alpha_{2P}(r_j, T_j)\} = \{(r_j^{e_{21}} * T_j), \cdots, (r_j^{e_{2P}} * T_j)\}$ using her secret blinding factor r_j and

authorities' public keys $e_{(1*)} = \{e_{(11)}, \cdots, e_{(1P)}\}$ and $e_{(2*)} = \{e_{(21)}, \cdots, e_{(2P)}\}$, respectively. While confirming the identity of V_j by anonymous tag based credential i.e., $T_j(A, ID_j, Z_j)$ of V_j, authorities $TM_1, \cdots, TM_P$ blindly sign on $\alpha_{*1}(r_j, T_j)$ and $\alpha_{*2}(r_j, T_j)$ to generate 2 different sets i.e., $t(d_{(1*)}, \alpha_{*1}(r_j, T_j)) = \{t(d_{(11)}, \alpha_{11}(r_j, T_j)), \cdots, t(d_{(1P)}, \alpha_{1P}(r_j, T_j))\} = \{\alpha_{11}(r_j, T_j)^{d_{11}}, \cdots, \alpha_{1P}(r_j, T_j)^{d_{1P}}\}$ and $t(d_{(2*)}, \alpha_{*2}(r_j, T_j)) = \{t(d_{(21)}, \alpha_{21}(r_j, T_j)), \cdots, t(d_{(2P)}, \alpha_{2P}(r_j, T_j))\} = \{\alpha_{21}(r_j, T_j)^{d_{21}}, \cdots, \alpha_{2P}(r_j, T_j)^{d_{2P}}\}$ by using their secret signing keys $d_{(1*)} = \{d_{(11)}, \cdots, d_{(1P)}\}$ and $d_{(2*)} = \{d_{(21)}, \cdots, d_{(2P)}\}$, respectively. Now V_j unblinds them into 2 unblinded signed tokens i.e., $s(d_{(1*)}, T_j) = \{s(d_{(11)}, T_j), \cdots, s(d_{(1P)}, T_j)\} = \{(\alpha_{11}(r_j, T_j)^{d_{11}}) * r_j^{-1}, \cdots, (\alpha_{1P}(r_j, T_j)^{d_{1P}}) * r_j^{-1}\}$ and $s(d_{(2*)}, T_j) = \{s(d_{(21)}, T_j), \cdots, s(d_{(2P)}, T_j)\} = \{(\alpha_{21}(r_j, T_j)^{d_{21}}) * r_j^{-1}, \cdots, (\alpha_{2P}(r_j, T_j)^{d_{2P}}) * r_j^{-1}\}$. Then, because authorities TMs have signed without knowing T_j, no one except V_j can know V_j from $s(d_{(1*)}, T_j)$ and $s(d_{(2*)}, T_j)$.

3.4 Signature Pairs on Blinded Vote

In vote submission stage the voter V_j uses her secret blinding factors (r_{1j}, r_{2j}), a pair of primes (a_{1j}, a_{2j}) and 1st form of public keys $e'_{(1*)} = \{e'_{(11)}, \cdots, e'_{(1P)}\}$ of authorities $TM_1, \cdots, TM_P$ to blind her vote v_j in the 1st form *i.e.*, V_j calculates $\alpha_{1*j} = \{(\alpha_{111j}, \alpha_{211j}), \cdots, (\alpha_{11Pj}, \alpha_{21Pj})\} = \{\{(r_{1j}{}^{e'11} * v_j{}^{a1j}), (r_{2j}{}^{e'11} * v_j{}^{a2\ j})\}, \cdots, \{(r_{1j}{}^{e'1P} * v_j{}^{a1j}), (r_{2j}{}^{e'1P} * v_j{}^{a2j})\}\}$. Similarly using 2nd form of public keys $e'_{(2*)} = \{e'_{(21)}, \cdots, e'_{(2P)}\}$ of $TM_1, \cdots, TM_P$, V_j blinds her v_j in the 2nd form *i.e.*, calculates $\alpha_{2*j} = \{(\alpha_{121j}, \alpha_{221j}), \cdots, (\alpha_{12Pj}, \alpha_{22Pj})\} = \{\{(r_{1j}{}^{e'21} * v_j{}^{a1j}), (r_{2j}{}^{e'21} * v_j{}^{a2j})\}, \cdots, \{(r_{1j}{}^{e'2P} * v_j{}^{a1j}), (r_{2j}{}^{e'2P} * v_j{}^{a2j})\}\}$. Here V_j blinds her vote *i.e.*, calculates $(\alpha_{1*j}, \alpha_{2*j})$ using individual public keys of independent authorities. Now authorities $TM_1, \cdots, TM_P$ sign on $(\alpha_{1*j}, \alpha_{2*j})$ using their 2 different sets of signing keys to generate 2 different forms of blinded signed vote. The 1st form of blinded signed vote is calculated as $t_1(d'_{(1*)}, (\alpha_{1*j}, \alpha_{2*j})) = \{(t_{111}, t_{211}), \cdots, (t_{11P}, t_{21P})\} = \Big\{\{(\alpha_{111j}{}^{b11d'11}), (\alpha_{211j}{}^{b21d'11})\}, \cdots, \{(\alpha_{11Pj}{}^{b1Pd'1P}), (\alpha_{21Pj}{}^{b2Pd'1P})\}\Big\}$. Similarly the 2nd form of blinded signed vote is calculated as $t_2(d'_{(2*)}, (\alpha_{1*j}, \alpha_{2*j})) = \{(t_{121}, t_{221}), \cdots, (t_{12P}, t_{22P})\} = \Big\{\{(\alpha_{121j}{}^{b'11d'21}), (\alpha_{221j}{}^{b'21d'21})\}, \cdots, \{(\alpha_{12Pj}{}^{b'1Pd'2P}), (\alpha_{22Pj}{}^{b'2Pd'2P})\}\Big\}$. Here 2 forms of blinded signed vote *i.e.*, $t_1(d'_{(1*)}, (\alpha_{1*j}, \alpha_{2*j}))$ and $t_2(d'_{(2*)}, (\alpha_{1*j}, \alpha_{2*j}))\}$ are generated by using the pair of signing keys $(d'_{(1*)}, d'_{(2*)})$ and 2 pairs of primes $\{(b_{(1*)}, b_{(2*)}), (b'_{(1*)}, b'_{(2*)})\}$ of $TM_1, \cdots, TM_P$ respectively; where $d'_{(1*)} = \{d'_{(11)}, \cdots, d'_{(1P)}\}$, $d'_{(2*)} = \{d'_{(21)}, \cdots, d'_{(2P)}\}$ and $b_{(1*)} = \{b_{(11)}, \cdots, b_{(1P)}\}$, $b_{(2*)} = \{b_{(21)}, \cdots, b_{(2P)}\}$, $b'_{(1*)} = \{b'_{(11)}, \cdots, b'_{(1P)}\}$, $b'_{(2*)} = \{b'_{(21)}, \cdots, b'_{(2P)}\}$. Now V_j generates 2 forms of unblinded signed vote from her blinded signed vote *i.e.*, calculates the 1st form $s_1(d'_{(1*)}, v_{j*}) = \{\{((v_j^{a_1jb_{11}d'_{11}})^{w_{11}}) \times ((v_j^{a_2jb_{11}d'_{11}})^{u_{11}})\}, \cdots, \{((v_j^{a_1jb_1Pd'_1P})^{w_1P}) \times ((v_j^{a_2jb_2Pd'_1P})^{u_1P})\}\}$ and the 2nd form $s_2(d'_{(2*)}, v_{j*}) = \{\{((v_j^{a_1jb_{21}d'_{21}})^{w_{21}}) \times ((v_j^{a_2jb_{21}d'_{21}})^{u_{21}})\}, \cdots, \{((v_j^{a_1jb_1Pd'_2P})^{w_2P}) \times ((v_j^{a_2jb_2Pd'_2P})^{u_2P})\}\}$. Herein for convenience, the signature derivation of the unblinding phase of Hwang *et al.*'s BS is directly shown where $\Big\{\{(w_{11j}, \cdots, w_{1Pj}), (u_{11j}, \cdots, u_{1Pj})\}$ and $\{(w_{21j}, \cdots, w_{2Pj}), (u_{21j}, \cdots, u_{2Pj})\}\Big\}$ are 2 pairs of integers of V_j. When each authority TM_i signs on $(\alpha_{1*j}, \alpha_{2*j})$ by his 2 different signing keys, it is impossible for any other entity to consistently generate 2 different signed forms *i.e.*, $\{t_1(d'_{(1*)}, (\alpha_{1*j}, \alpha_{2*j})), t_2(d'_{(2*)}, (\alpha_{1*j}, \alpha_{2*j}))\}$ in an unauthorized way because each TM_i knows only his secret signing key. V_j can convince herself that TMs have signed on $(\alpha_{1*j}, \alpha_{2*j})$ honestly when she unblinds $\{t_1(d'_{(1*)}, (\alpha_{1*j}, \alpha_{2*j})), t_2(d'_{(2*)}, (\alpha_{1*j}, \alpha_{2*j}))\}$ to $\{s_1(d'_{(1*)}, v_{j*}), s_2(d'_{(2*)}, v_{j*})\}$ and verifies the signatures.

3.5 Anonymous Tag Based Credential

Anonymous tag based credential $T_j(A, ID_j, Z_j)$ proposed in [33] provided by the credential issuer A enables a voter V_j to prove her eligibility to any entity e.g. voting manager VM without revealing her identity where ID_j and Z_j is the identity and a secret random integer of V_j. Here initially V_j shows her identity to A, then A gives the credential $T_j(A, ID_j, Z_j)$ to V_j if she is eligible. Later on, any entity including VM can force V_j to calculate the used seal $U_j^{Z_j}$ $(mod\ n)$ from a given integer U_j while using Z_j in $T_j(A, ID_j, Z_j)$ honestly without knowing Z_j himself. Here n is a large and appropriate public integer associated with $T_j(A, ID_j, Z_j)$ and in the following, the notation $(mod\ n)$ is omitted. Then, any entity like VM can use $U_j^{Z_j}$ as an evidence that V_j had shown $T_j(A, ID_j, Z_j)$ to him. In conclusion, together with the used seal $U_j^{Z_j}$ anonymous credential $T_j(A, ID_j, Z_j)$ satisfies anonymity, unlinkability, verifiability, unforgeability, soundness, and revocability [29, 33].

4 Configuration of the Scheme

The proposed scheme consists of N voters $V_j (j = 1, \cdots, N)$ where j means j-th voter, a single (or multiple) Voting manger VM, P mutually independent Tallying managers TM_i $(i = 1, \cdots, P)$ where P is at least 2, Credential issuer A and four bulletin boards (BBs) [17] namely, *VoterList, TokenList, VotingBoard* and *TallyingBoard*. Figure 1 shows the configurations of each BB at some arbitrary point during the election. Here all the relevant information of interactions among the entities at every stage of the election are posted on BBs, thereby the scheme becomes publicly verifiable. Roles of the above

ID	credential	blinded token
ID_1	$T_1(A, ID_1, Z_1)$	$\alpha_{*1}(r_1, T_2)$, $\alpha_{*2}(r_1, T_2)$
. . .	. . .	. . .
ID_j	$T_j(A, ID_j, Z_j)$	$\alpha_{*1}(r_j, T_j)$, $\alpha_{*2}(r_j, T_j)$
. . .	. . .	. . .
ID_N	$T_N(A, ID_N, Z_N)$	$\alpha_{*1}(r_N, T_8)$, $\alpha_{*2}(r_N, T_8)$

(a) VoterList

token	seal
T_1	-
. . .	. . .
T_j	U_j^{Zj}
. . .	. . .
T_N	U_N^{ZN}

(b) TokenList

blinded vote	approval
$\{t_1(d'_{(1*)}, (\alpha_{1*q}, \alpha_{2*q})), t_2(d'_{(2*)}, (\alpha_{1*q}, \alpha_{2*q}))\}$	$s(d_{(1*)}, T_{11})$
. . .	. . .
$\{t_1(d'_{(1*)}, (\alpha_{1*j}, \alpha_{2*j})), t_2(d'_{(2*)}, (\alpha_{1*j}, \alpha_{2*j}))\}$	$s(d_{(1*)}, T_j)$
. . .	. . .
$\{t_1(d'_{(1*)}, (\alpha_{1*c}, \alpha_{2*c})), t_2(d'_{(2*)}, (\alpha_{1*c}, \alpha_{2*c}))\}$	$s(d_{(1*)}, T_N)$

(c) VotingBoard

unblinded vote	approval
$s_1(d'_{(1*)}, v_{q*}), s_2(d'_{(2*)}, v_{q*})$	$s(d_{(2*)}, T_{11})$
. . .	. . .
$s_1(d'_{(1*)}, v_{j*}), s_2(d'_{(2*)}, v_{j*})$	$s(d_{(2*)}, T_j)$
. . .	. . .
$s_1(d'_{(1*)}, v_{c*}), s_2(d'_{(2*)}, v_{c*})$	$s(d_{(2*)}, T_N)$

(d) TallyingBoard

Figure 1: Configuration of bulletin boards

mentioned entities are as follows:

Voter V_j: Each voter V_j has her own ID_j and P/W_j to prove her eligibility to the credential issuer A while obtaining anonymous credential $T_j(A, ID_j, Z_j)$ from him. V_j uses seal $U_j^{Z_j}$ to approve the acquisition of unused token T_j, and secret blinding factor r_j to blind her token T_j to $\{\alpha_{*1}(r_j, T_j)$ and $\alpha_{*2}(r_j, T_j)\}$. She also has a pair of blinding factor $\{r_{1j}, r_{2j}\}$, a pair of primes $\{a_{1j}, a_{2j}\}$ and another 2 pairs of integers $\Big\{\{(w_{11j}, \cdots, w_{1Pj}), (u_{11j}, \cdots, u_{1Pj})\}$ and $\{(w_{21j}, \cdots, w_{2Pj}), (u_{21j}, \cdots, u_{2Pj})\}\Big\}$ to blind and unblind her vote v_j.

Voting manager *VM*: *VM* verifies V_j's eligibility anonymously using $T_j(A, ID_j, Z_j)$, puts voter's seal $U_j^{Z_j}$ on TokenList, blinded votes on *VotingBoard* and maintains *VoterList*, and *TallyingBoard* by putting data about voters, tokens and unblinded votes. *VM* also signs on each T_j prior to post on *TokenList*. If necessary multiple independent *VM* can be constructed for distributing its responsibility and achieving more reliability.

Tallying managers *TMs*: There are $P(P \geq 2)$ mutually independent TMs. Each TM_i has the responsibility to sign on blinded token $\{\alpha_{*1}(r_j, T_j)$ and $\alpha_{*2}(r_j, T_j)\}$ and blinded vote $(\alpha_{1*j}, \alpha_{2*j})$ with his 2 different forms of signing keys. TM_i has a pair of signing keys $\{d_{(1i)}, d_{(2i)}\}$ to sign on blinded token $\Big\{\alpha_{*1}(r_j, T_j)$ and$\alpha_{*2}(r_j, T_j)\Big\}$ in 2 different forms. To sign on a blinded vote he has a pair of signing keys $\{d'_{(1i)}, d'_{(2i)}\}$ and another 2 pairs of primes $\Big\{b_{(1i)}, b_{(2i)}\Big\}$, $\{b'_{(1i)}, b'_{(2i)}\}$. Here each signing key has its corresponding public key.

Credential issuer A: A is responsible to generate and issue an anonymous tag based credential $T_j(A, ID_j, Z_j)$ to each V_j.

VoterList: 3 parts named ID, credential and token parts form *VoterList*. ID part contains the ID_j of eligible V_j, credential part contains anonymous credential $T_j(A, ID_j, Z_j)$ and token part contains the blinded form of token i.e., $\Big\{\alpha_{*1}(r_j, T_j)$ and $\alpha_{*2}(r_j, T_j)\Big\}$ of its corresponding voter's ID as shown in Figure 1 (a). As this is a BB, anyone can monitor the list.

TokenList: *TokenList* consists of the token and seal parts, and permits an anonymous V_j to acquire T_j without collision. The token part maintains tokens i.e., unique numbers already prepared by *VM*. Through anonymous credential [33] while voter V_j picks a token T_j, *VM* puts V_j's seal $U_j^{Z_j}$ on seal part of *TokenList* as shown in Figure 1 (b).

VotingBoard: *VotingBoard* consists of the blinded vote and the approval part. Blinded vote part at t_j-th position contains 2 different forms of blinded signed vote of the voter to whom t_j-th token T_j is assigned. So, vote part consists of TMs' 1st and 2nd forms of signatures on blinded vote i.e., $t_1\Big(d'_{(1*)}, (\alpha_{1*j}, \alpha_{2*j})\Big)$ and $t_2\Big(d'_{(2*)}, (\alpha_{1*j}, \alpha_{2*j})\Big)$. Approval part contains the 1st form of unblinded signed T_j i.e., $s(d_{(1*)}, T_j)$ that approves the vote of V_j on *VotingBoard* as shown in Figure 1(c).

TallyingBoard: *TallyingBoard* contains an unblinded vote part and an approval part. Unblinded vote part contains the vote unblinded by its voter in 2 different signed forms i.e., $s_1(d'_{(1*)}, v_{j*})$ and $s_2(d'_{(2*)}, v_{j*})$. V_j approves the correctness of *TMs* signatures on her unblinded vote by putting the 2nd form of unblinded signed T_j i.e., $s(d_{(2*)}, T_j))$ signed by *TMs* on the approval part of TallyingBoard as shown in Figure 1 (d). Anyone can monitor voters who have unblinded and approved their votes.

5 Overview of the Scheme

The proposed scheme consists of 4 stages and this section briefly describes them as follows. Figure 2 represents the relationships and the data flows among entities involved in the stages of the scheme.

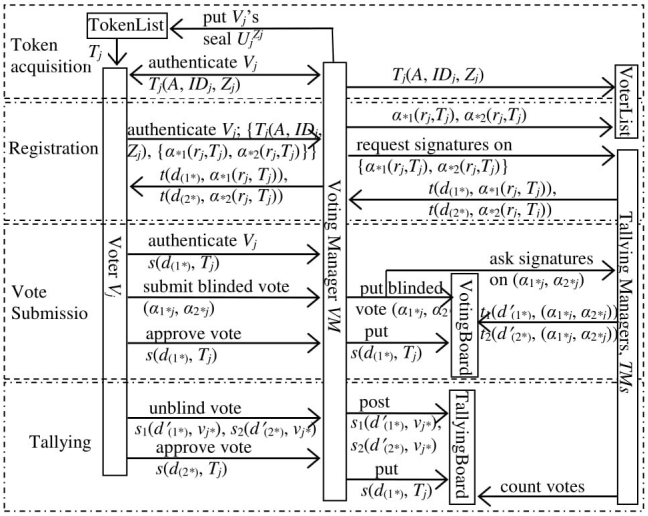


Figure 2: Relationships and data flow among entities of the scheme

5.1 Token Acquisition

Using anonymous credential $T_j(A, ID_j, Z_j)$ and seal $U_j^{Z_j}$, each anonymously authenticated voter V_j picks an unused token T_j from *TokenList*.

5.2 Registration

Voter V_j gets herself authenticated using credential $T_j(A, ID_j, Z_j)$. Then V_j submits her blinded token T_j i.e., $\left\{\alpha_{*1}(r_j, T_j) \text{ and } \alpha_{*2}(r_j, T_j)\right\}$ to *VM* to post it on *VoterList*. V_j gets 2 kinds of signatures of TMs on blinded T_j i.e., $t(d_{(1*)}, \alpha_{*1}(r_j, T_j))$ and $t(d_{(2*)}, \alpha_{*2}(r_j, T_j))$. These 2 forms of signed T_j help V_j to prove her eligibility in further stages. 1st form of unblinded signed T_j i.e., $s(d_{(1*)}, T_j)$ is used to approve V_j's vote on VotingBoard and 2nd form of unblinded signed T_j i.e., $s(d_{(2*)}, T_j))$ is used to approve V_j's unblinded signed vote on *TallyingBoard*.

5.3 Vote Submission

Employing Hwang *et al.*'s BS, V_j calculates $(\alpha_{1*j}, \alpha_{2*j})$ as her blinded vote as described in Section 3.4 and submits it along with $s(d_{(1*)}, T_j)$ to *VM*, to put $(\alpha_{1*j}, \alpha_{2*j})$ on *VotingBoard*. TMs sign on it by their 1st and 2nd form of signing keys i.e., produce $t_1(d'_{(1*)}, (\alpha_{1*j}, \alpha_{2*j}))$ and $t_2(d'_{(2*)}, (\alpha_{1*j}, \alpha_{2*j}))$. While checking her blinded vote on *VotingBoard*, V_j approves it by putting $s(d_{(1*)}, T_j)$ on the approval part of *VotingBoard*.

5.4 Tallying

While vote submission ends, every V_j unblinds her blinded signed vote by calculating $s_1(d'_{(1*)}, v_{j*})$ and $s_2(d'_{(2*)}, v_{j*})$ as discussed in Section 3.4. V_j checks the correctness of TMs' signatures and submits $s_1(d'_{(1*)}, v_{j*})$ and $s_2(d'_{(2*)}, v_{j*})$ to *VM* to be posted on *TallyingBoard*. Also by putting $s(d_{(2*)}, T_j)$ on the approval part of *TallyingBoard*, V_j approves her unblinded signed vote.

6 Individual Stages of the Scheme

The stages of the scheme proceed as follows.

6.1 Token Acquisition Stage

In this stage each voter V_j acquires a token T_j which is unique in the system, while maintaining the anonymity of V_j. For this purpose, at least N pre-generated tokens are put in *TokenList* from where a voter picks her token without collisions; where N is the number of eligible voters. Every T_j of *TokenList* has the signature of *VM* (this signature is different from $s(d_{(1*)}, T_j)$ and $s(d_{(2*)}, T_j)$, and ensures that T_j has been picked from *TokenList*). The authentication of V_j in this stage is not so essential. But the use of anonymous credential $T_j(A, ID_j, Z_j)$ protects T_j from being picked by unauthorized entities; and thereby *TokenList* remains as small as possible. During this stage V_j and *VM* interacts as follows:

1) *VM* anonymously authenticates eligible voter V_j by anonymous tag based credential [33].

2) After authentication, *VM* updates *VoterList* by putting $T_j(A, ID_j, Z_j)$ as shown in Figure 1(a).

3) Authenticated V_j picks an unused token T_j form *TokenList*, and *VM* puts his signature on the T_j (although this notation of signature is omitted in this paper). Now V_j submits her seal $U_j^{Z_j}$ to *VM*.

4) As T_j has been picked up by V_j, *VM* puts the seal $U_j^{Z_j}$ of V_j corresponding to it on *TokenList* as shown in Figure 1(b).

Security issues of this stage are as follows:

- *Single V_j may get multiple tokens:* *VM* puts the seal $U_j^{Z_j}$ of V_j corresponding to her T_j on *TokenList* in exchange of the credential. Therefore V_j cannot request multiple tokens.

- *A voter may not get a token:* As at least N tokens are generated, every voter gets a token. If any V_j cannot get a token, she can request repeatedly.

- A voter may use her own token: On T_j to get the signatures of *TMs*, *VM* accepts a token that has his (*VM*) signature. Therefore V_j cannot use her own T_j.

6.2 Registration Stage

Tallying managers *TMs* sign on 2 different forms of blinded T_j, i.e., $\alpha_{*1}(r_j, T_j)$ and $\alpha_{*2}(r_j, T_j)$ of Vj during this stage, inside of the voting booth. Firstly voter V_j

blinds her T_j in 2 different forms and then *TMs* blindly sign on them as described in Section 3.3, so that *TMs* sign on T_j without knowing its' content. This signed blinded T_j proves the eligibility of V_j anonymously in later stages. *VM* maintains *VoterList* as shown in Figure 1 (a) showing registered voter's *ID*, each voter's credential $T_j(A, ID_j, Z_j)$ and blinded T_j. As *VoterList* is public, anyone can monitor a registered V_j without knowing T_j as T_j on *VoterList* is in blinded form, i.e., $\{\alpha_{*1}(r_j, T_j)$ and $\alpha_{*2}(r_j, T_j)\}$. In this stage V_j and *VM* interacts as follows:

1) V_j blinds her token T_j in 2 different forms as $\{\alpha_{*1}(r_j, T_j)$ and $\alpha_{*2}(r_j, T_j)\}$using her secret blinding factor r_j.

2) V_j shows her credential $T_j(A, ID_j, Z_j)$ and blinded token $\{\alpha_{*1}(r_j, T_j)$ and $\alpha_{*2}(r_j, T_j)\}$ to *VM*.

3) After authentication, *VM* updates *VoterList* by putting $\{\alpha_{*1}(r_j, T_j)$ and $\alpha_{*2}(r_j, T_j)\}$ as shown in Figure 1(a). *VM* also sends $\{\alpha_{*1}(r_j, T_j)$ and $\alpha_{*2}(r_j, T_j)\}$ to mutually independent *TMs* for their signatures.

4) $TM_1, \cdots, TM_P$ sign on $\{\alpha_{*1}(r_j, T_j)$ and $\alpha_{*2}(r_j, T_j)\}$ to generate 2 different forms *i.e.*, calculate $t(d_{(1*)}, \alpha_{*1}(r_j, T_j))$ and $t(d_{(2*)}, \alpha_{*2}(r_j, T_j))$ and sends them to *VM* to be sent to V_j.

5) V_j checks the validity of signatures on blinded T_j.

Security issues of this stage are as follows:

- *VM may misuse signed T_j:* This security issue can arise if single *VM* is engaged and he gets corrupted. To avoid the issue, multiple *VM* can be employed. Thereby unless all *VM*s get corrupted, signatures of all *TMs* cannot be collected on T_j.
- *VM may put invalid signature on blinded T_j:* V_j can prove *VM*'s dishonesty by showing $\{\alpha_{*1}(r_j, T_j)$ and $\alpha_{*2}(r_j, T_j)\}$ and the incorrect signed token.
- *Signed token T_j may be given to a coercer:* If signed T_j is stolen, V_j is responsible for that. However for voting while V_j comes to a voting booth, she cannot interact with an external coercer. Authorities *e.g. VM* or *TMs* cannot coerce a voter unless all of them get corrupted.

6.3 Vote Submission Stage

V_j uses her 1st form of unblinded signed token *i.e.*, $s(d_{(1*)}, T_j)$ to be authenticated. *VM* checks V_j's validity by verifying the signatures of *TMs* on T_j *i.e.*, $s(d_{(1*)}, T_j)$. Then V_j blinds her vote v_j in 2 different forms by using blinding factors (r_{1j}, r_{2j}), primes (a_{1j}, a_{2j}) and *TMs*' public keys (e'_{1*}, e'_{2*}) by calculating $(\alpha_{1*j}, \alpha_{2*j})$ as described in Section 3.4 *i.e.*, 2 forms of blinded vote of V_j are

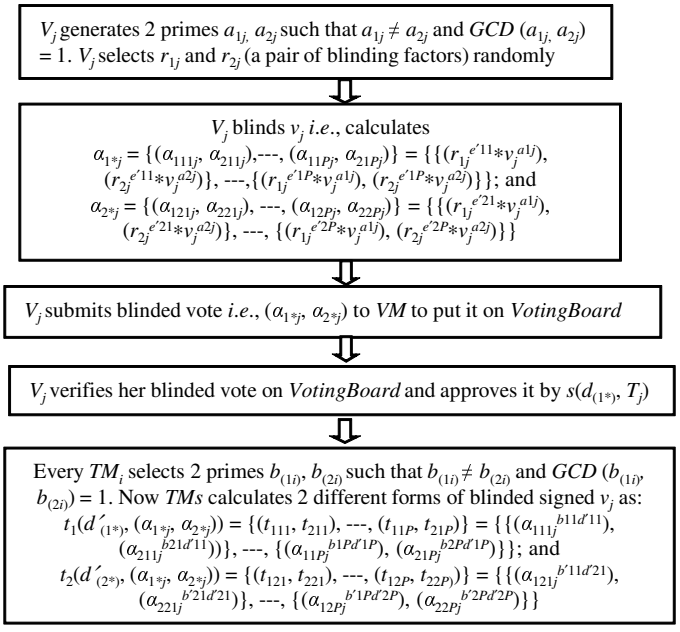


Figure 3: Vote construction procedure

$(\alpha_{1*j}, \alpha_{2*j})$. Now V_j sends $(\alpha_{1*j}, \alpha_{2*j})$ to *VM* to put on *VotingBoard*. After finding her blinded vote on *VotingBoard*, V_j approves it by sending $s(d_{(1*)}, T_j)$ to *VM* to be posted on the approval part of *VotingBoard*. Therefore anyone can monitor a voter who has submitted her blinded vote without knowing her identity and the actual vote. Finally *TM*s sign on the blinded vote to be put on *VotingBoard* with their 1st and 2nd forms of signatures *i.e.*, calculate $t_1(d'_{(1*)}, (\alpha_{1*j}, \alpha_{2*j}))$ and $t_2(d'_{(2*)}, (\alpha_{1*j}, \alpha_{2*j}))$ as described in Section 3.4. The vote construction procedure is shown in Figure 3. Steps of this stage are as follows:

1) V_j submits $s(d_{(1*)}, T_j)$ to *VM*. By checking only the validity of signatures on T_j that is not repeatedly used, *VM* checks the validity of V_j.

2) V_j blinds her vote v_j *i.e.*, calculates $(\alpha_{1*j}, \alpha_{2*j})$ as discussed in Section 3.4.

3) V_j submits $(\alpha_{1*j}, \alpha_{2*j})$ as blinded vote to *VM* to post it on *VotingBoard* (however, it is not shown on *VotingBoard*).

4) By checking her blinded vote on *VotingBoard*, V_j approves it by sending $s(d_{(1*)}, T_j)$ to be posted on *VotingBoard* also.

5) $TM_1, \cdots, TM_P$ sign on the blinded vote $(\alpha_{1*j}, \alpha_{2*j})$ on *VotingBoard* with their 1st and 2nd form of signatures *i.e.*, calculate $t_1(d'_{(1*)}, (\alpha_{1*j}, \alpha_{2*j}))$ and $t_2(d'_{(2*)}, (\alpha_{1*j}, \alpha_{2*j}))$ as discussed in Section 3.4 and post them on *VotingBoard* as shown in Figure 1(c).

For this stage the security issues are as follows:

- *Voter may submit invalid vote to disrupt voting:* V_j herself submits and approves her blinded signed vote

in 2 different forms on *VotingBoard*. Later on, V_j cannot claim that her vote is disrupted even if the vote is meaningless when unblinded vote in 2 signed forms *i.e.*, $s_1(d'_{(1*)}, v_{q*})$ and $s_2(d'_{(2*)}, v_{q*})$ are consistent.

- *VM may not put vote or put incorrect vote:* As *VoterList* is open to the public, repeatedly the V_j can ask *VM* to put her vote on *VotingBoard* by submitting the vote before her approval. If *VM* puts incorrect vote on *VotingBoard*, V_j can disapprove it.
- *Votes in VotingBoard can be modified by attacker:* As *VotingBoard* is open to the public, no one can modify its contents illegally.

6.4 Tallying Stage

All votes on *VotingBoard* are in blinded form. When vote submission ends, each voter needs to unblind her vote in 2 different signed forms *i.e.*, calculates $s_1(d'_{(1*)}, v_{j*})$ and $s_2(d'_{(2*)}, v_{j*})$ as described in Section 3.4. V_j checks the correctness of *TMs*' signatures on her blinded vote. Now V_j submits $s_1(d'_{(1*)}, v_{j*})$ and $s_2(d'_{(2*)}, v_{j*})$ to *VM* to put it on *TallyingBoard*. Then, V_j approves them by posting 2nd form of her signed T_j *i.e.*, $s(d_{(2*)}, T_j)$ on the approval part of *TallyingBoard*. Here V_j's data on *VotingBoard* and *TallyingBoard* may be corresponding or not. If corresponding, easily it is seen that the same blinded and unblinded signed vote on 2 *BBs* is approved by the same T_j. If not corresponding and no approval is put on *TallyingBoard*, no one including *TMs* can know the link between them because of Hwang *et al.*'s BS. Thus links among blinded signed vote on *VotingBoard*, unblinded signed vote on *TallyingBoard* and the identity of a registered V_j on *VoterList* is removed. Steps of this stage are as follows:

1) V_j unblinds her 2 forms of blinded signed vote as $\{s_1(d'_{(1*)}, v_{j*}), s_2(d'_{(2*)}, v_{j*})\}$ and checks the correctness of *TMs*' signatures on them.
2) V_j submits $s_1(d'_{(1*)}, v_{j*})$ and $s_2(d'_{(2*)}, v_{j*})$ to *VM* to post them on *TallyingBoard*.
3) By sending 2nd form of her unblinded signed T_j, *i.e.*, $s(d_{(2*)}, T_j)$ to *VM* to put it on the approval part of *TallyingBoard*, V_j approves her vote.

Security issues of this stage are as follows:

- *Voter may not unblind her vote:* If V_j does not unblind her vote, the vote cannot be considered for counting. However, it is obvious in any application of BS that the entity that blinds the data must unblinds it.
- *TMs may add or delete votes:* By this the numbers of votes on *VotingBoard* and *TallyingBoard* become different which is detectable by anyone.

7 Performance Analysis

This section evaluates the prototype of the proposed scheme and compares it with other schemes.

7.1 Experiment Setup

To measure the computation time requirement for Registration, Voting and Tallying stages, a prototype of the proposed scheme consists of 3 independent Tallying mangers is developed i.e. no client-server based web application is developed in a realistic environment where multiple entities are distributed over different places. Therefore all computation times do not include the communication time. The prototype is developed under the environment of Intel Core i3-3.10 GHz processor with 4 GBytes of RAM running on Windows 7 operating system. For cryptographic operations, GMP [12] with 1024 bit and 2048 bit modulus has been used. Besides, it is assumed that blinding factors, secret integers, primes, *etc.* of involved entities are prepared in advance. Also, operations of entities that are not related to cryptography are not considered.

7.2 Performance Evaluation

Table 2: Time requirement for registration, vote submission and tallying stages

	Stages (time in ms)					
Phase	Registration		Vote Submission		Tallying	
	1024 bit	2048bit	1024 bit	2048 bit	1024 bit	2048 bit
Blinding	0.216	0.804	4.740	17.136	–	–
Signing	3.672	23.130	26.220	179.898	–	–
Unblinding	0.018	0.072	–	–	11.322	40.374
Verification	–		–	–	0.234	0.888
Total	3.906	24.006	30.94	197.034	11.556	41.262

During Registration stage V_j blinds her token T_j in 2 different forms, *TMs* sign on them and V_j unblinds them to obtain unblinded signed T_j. As there are 3 *TMs*, V_j blinds her T_j in 6 forms, blinded T_j is signed by *TMs* and 6 forms are generated, and finally V_j unblinds them all. Vote submission stage consists of blinding the vote v_j in 2 different forms and signing on them. Because of 3 *TMs*, V_j blinds her v_j in 6 forms by using 2 different public keys of 3 *TMs*, and blinded v_j is signed by *TMs* and 6 forms are generated. In Tallying stage, voter V_j unblinds her blinded signed vote v_j in 6 forms and finally anyone can verify the vote. The time requirement for different operations in Registration, Vote submission and Tallying stages for the proposed scheme using GMP with 1024 bit and 2048 bit modulus has been summarized in Table 2. Using GMP the total time requirement for Registration, Vote submission and Tallying stages are 3.906ms, 30.94ms, and 11.556ms respectively for 1024 bit; while for 2048 bit it requires 24.006ms, 197.034ms and 41.262ms respectively.

7.3 Discussions

For unblinding any data using both Hwang *et al.*'s BS and Chaum's BS, equations that have the form like: $s = t \cdot r^{-b} \bmod n$ — (1), are solved by using Extended Euclidean Algorithm [31] that finds out x and y when $ax + by = GCD(a, b)$. If $GCD(a, b)$ is 1, then $ax + by = 1$. Equation (1) can be rewritten as $r^b . s = ny + t$ where y is a positive integer. Hence, equation (1) becomes $r^b .(s/t) + n .(- y/t) = 1$. Now the value of (s/t) and $(- y/t)$ can be found by using Extended Euclidean Algorithm. As t is known, s can be easily calculated. The operations involved in unblinding phase of both schemes have been evaluated in this way. Chinese Remainder Theorem [31] is used to evaluate the signing phase of Chaum's BS that has shrunk the computation time of this phase.

The computation time requirement for blinding tokens and votes, signing on blinded tokens and votes and unblinding signed tokens and votes are directly proportional to the numbers of *TMs* involved in the scheme. Using GMP with 1024 bit modulus, 1000 votes can be counted within 12 seconds (0.011556 * 1000 = 11.556) which is feasible enough to implement in real world. To get an overview of the proposed scheme if 100 thousand voters (0.1 million) are considered using 1024 bit modulus implemented with GMP; the Registration, Vote Submission and Tallying stages can be completed within 78 minutes on a single server (*i.e.*, (0.046402 * 100000) = (4640.2secs/ 60) = 77.34 min).

7.4 Comparisons

Table 3: Computation time comparisons with other schemes

Schemes	CPU(GHz)	Memory	Coding	1024 bit modulus (time in ms)		
				Registration	Voting	Tallying
Proposed scheme	3.10	4 GB	GMP	3.906	30.94	11.556
CNSc	1.60	504 MB	GMP	47.1	308	171
DynaVote	1.60	752MB	Java	-	2470	208.3

The performance of prototype of the proposed scheme is compared with those of confirmation number (CN) based anonymous voting scheme (CNSc) proposed in [3], and DynaVote proposed in [6] which are available for comparisons, although the used hardware configurations and coding platforms are not same. Thereby the comparison is not an absolute one. Also, no comparison with schemes that deploy ZKP, e.g., Helios [2] Civitas [9] has been presented (a comparison with a ZKP based scheme is available in [3]) because for ZKP it requires huge computation time. Moreover, no comparison with schemes that allow the same voter to cast her vote multiple times, e.g., UVote [1] has been made because the proposed scheme does not consider the vote submission in this way. In CNSc [3], the voter's Registration stage is identical to the proposed scheme. In Voting stage, the vote construct consists of: i) the voter encrypts her vote, ii) 3 authorities' perform triple encryptions on it, iii) the voter decrypts it by her decryption key, iv) the voter verifies authorities' encryptions of vote, v) 3 authorities repeatedly sign on the encrypted vote in 2 different forms and on the confirmation number in a single form, and finally vi) the voter verifies both forms of authorities' signatures. The time requirement for tallying is comprised of decryptions and shuffles and verifications of 2 signed forms of votes and single signed form of CNs. In DynaVote [6] the prototype has been developed over the internet, and while considering 1000 votes the runtime requirement of each vote in Voting stage is 2470.042ms and in Tallying stage is 208.3ms. Although the communication between server and client uses multi-threading, it did not use this feature while testing the prototype. Here voting stage consists of ballot obtaining and vote casting phases, and while the number of votes increases, the time requirement decreases gradually. A comparison among the schemes for a single vote and its voter has been presented in Table 3.

7.5 Untraceability

The proposed scheme maintains the untraceability property of Hwang *et al.*'s BS referring to the fact that for any given valid signature $\{v_j, s(d'_{(i)}, v_j)\}$, the authority TM_i is unable to link the signature to the vote. The demonstration is as follows. As described in Sections 6.3 and 3.4, the voter V_j submits her blinded vote *i.e.*, $(\alpha_{1ij}, \alpha_{2ij})$, and TM_i signs on it *i.e.*, calculates $t(d'_{(i)}, (\alpha_{1ij}, \alpha_{2ij}))$ using his primes (b_{1i}, b_{2i}). Now TM_i can store a set of records *i.e.*, $\{(\alpha_{1ij}, \alpha_{2ij}), t(d'_{(i)}, (\alpha_{1ij}, \alpha_{2ij})), (b_{1i}, b_{2i})\}$ for every blinded vote. During the Tallying stage when V_j reveals her unblinded signed vote as $\{s(d'_{(i)}, v_j)\}$ by putting it on *TallyingBoard*, TM_i has no way to get any information regarding V_j's secret blinding factor (r_{1j}, r_{2j}) from the stored information. Moreover, V_j's unblinded signed vote consists of two parts *i.e.*, $s(d'_{(i)}, v_j)$ has been generated from $\{(v_j{}^{a1jb1d'i})^{wj}\}$ and $\{(v_j{}^{a2jb2d'i})^{uj}\}$ (as discussed in Section 3.2) and neither of which TM_i knows. Hence without knowing V_j's secret blinding factor (r_{1j}, r_{2j}), pair of primes (a_{1j}, a_{2j}) and integers (w_j, u_j), TM_i cannot trace the BS. Here it is same for all authorities (*TMs*) while a vote v_j is constructed in any of 2 forms by any TM_i.

7.6 Further Extensions

An erasable-state voting booth as discussed in [29], can be deployed for the proposed scheme. Thereby, while the voter interacts with authorities, she is unable to memorize the complete list of information exchanged between herself and election authorities. For example, to construct her vote the voter uses lots of parameters like secret blinding factors, integers, primes *etc.* and later on she cannot reuse them. Thereby, she cannot prove her vote to any third party. Besides, the proposed scheme does not deploy

Table 4: Comparison of schemes based on security requirements

Schemes	Verifiable	Fair	Robust	Receipt-free	Accuracy	Dispute-free	Incoercible	Scalable	Practical	Major Tools
Proposed scheme	U	Y	Y	C	Y	Y	Y	Y	C	BS
Lee *et al.* [20]	U	C	C	Y	Y	N	N	C	N	Mixnet
CNSc [3]	U	Y	C	Y	Y	Y	Y	N	Y	HE, Mixnet
Fujioka *et al.* [11]	I	Y	N	N	N	N	N	Y	N	BS
Juang *et al.* [32]	I	C	C	N	C	N	N	Y	Y	BS
DynaVote [6]	I	C	NK	Y	Y	Y	Y	Y	C	BS
Helios [2]	Y	Y	Y	N	Y	N	N	N	Y	Mixnet, ZKP
Civitas [9]	I	Y	Y	Y	C	N	Y	N	N	Mixnet, ZKP
UVote [1]	I	Y	NK	N	Y	N	Y	NK	NK	Mixnet
Cobra [10]	N	Y	Y	Y	Y	NK	Y	N	N	HE

Y: Yes; N: No; NK: Not Known; I: Individually; U: Universally; C: Conditionally; P: Partially; BS: Blind Signature; HE: Homomorphic Encryption; ZKP: Zero Knowledge Proof;

any form of mixnet [15]. However, as discussed in [14]; a verifiable mixnet can also be incorporated herein. For this while vote submission, the voter submits her unblinded signed vote to the mixnet. When every voter completes her vote submission, the mixnet processes the encrypted votes i.e., either re-encrypts or decrypts and shuffles them. Finally an authority decrypts the votes shuffled by the mixnet and publishes the result on the BB. Herein, a little rearrangement of individual stages of the scheme will be required. Thereby, the scheme would become suitable for big community where the number of voters is high also.

8 Security Analysis

Based on major requirements, a comparison among the schemes has been presented in Table 4 where very basic requirements namely privacy, eligibility etc which are satisfied by almost schemes are omitted. But herein also, it is difficult to establish an absolute comparison because in many cases schemes cannot satisfy a particular requirement at the same level. Besides, the definition and the way of attaining requirements even may vary among schemes. For example to ensure un-reusability, some schemes assume that one voter can vote only once. But to attain incoercibility, many schemes enable one voter to cast her vote multiple times from which only a valid vote is counted. Also, there is tradeoffs among requirements. Therefore even by observing the Table, it is difficult to decide which particular scheme is the sole winner.

This section also discusses the way how the proposed scheme satisfies requirements of e-voting where their formal meanings are available in [14, 17, 24].

Privacy: By using 2 different forms of unblinded signed token, each voter submits as well as approves her vote anonymously. Thus, no one except the voter can know the link between blinded signed vote and its voter; and cannot identify a voter who did not submit her vote. Also the use of Hwang *et al.*'s BS disables entities even *TMs*' to link between blinded signed vote on *VotingBoard* and its unblinded signed form on *TallyingBoard* while they are not posted correspondingly, and the voter's approval does not appear on *TallyingBoard*.

Eligibility: While Token acquisition and Registration stages, the identity of the voter is identified by anonymous credential $T_j(A, ID_j, Z_j)$. Also to submit and approve the vote, the corresponding voter's identity is ensured by her unblinded signed T_j which is unique. Moreover, the token of each voter is signed by multiple authorities; therefore no one can forge signatures on T_j. Thus only eligible voters can participate in voting.

Un-reusability: While voter submits her vote using signed token, *VM* checks that the token is already used or not. Also the voter's blinded signed vote on *VotingBoard* and unblinded signed vote on *TallyingBoard* are approved by the same token only signed in 2 different forms; therefore multiple voting by a single voter is prevented.

Accuracy: Only unblinded signed votes approved by their voters appearing on *TallyingBoard* are considered for tallying. Thus all and only valid votes are counted.

Fairness: Every vote on *VotingBoard* is blinded by its corresponding voter and signed by all *TMs*; thereby no entity can know the interim voting results. Only the corresponding voter can unblind her vote during the Tallying stage.

Robustness: While even an invalid vote is identical within 2 unblinded signed forms, the voter cannot claim that her vote is disrupted; thus a voter can disrupt only her own vote. Also *VM* or *TMs* cannot disrupt the scheme if at least a single entity of them is honest among multiple entities.

Universal Verifiability: Every voter approves her blinded signed vote on *VotingBoard* and unblinded signed vote on *TallyingBoard* by her unique token signed in 2 different forms, which is publicly open. Moreover thereby, a registered voter can submit only a single vote. Thus the scheme ensures that all and only vote approved by its individual voter is counted.

Dispute-freeness: In the scheme, publicly-verifiable data about interactions among entities on different *BBs*, signature pairs on vote and signature pairs on unique token enable involved entities to resolve disputes.

Receipt-freeness: By deploying an erasable-state voting booth, receipt-freeness can be achieved. Due to an erasable-state voting booth, later on the voter cannot reuse her secret parameters to reconstruct the vote. Also as discussed in Section 6.3, the vote is constructed in distributed fashion through the involvement of the voter and *TMs*. Thereby, although the voter knows her blinded signed vote on *VotingBoard*, she cannot prove it to the coercer.

Incoercibility: When unblinded signed vote in 2 different forms are same, no one can claim that the vote is disrupted. Thus the scheme is free from randomization attack. Also a registered voter proves her identity to authorities anonymously through unique token signed by multiple authorities; therefore coercers cannot pretend to be a valid voter instead of herself. Thus the scheme is free from simulation attack.

Scalability: The scheme provides a scalable solution for major security aspects as discussed above. Also the prototype performance evaluation presented in Section 7 shows that the time requirement to implement the scheme is not so high.

Practicality: The scheme relies on an erasable-state voting booth to achieve receipt-freeness, although it is not yet implemented. Also herein, as BS is deployed for vote construction; obviously a voter needs to unblind her blinded signed vote later on. These impair the practicality. However, while the voter submits her unblinded signed vote to a mixnet as discussed in Section 7.6, the second problem is resolved.

9 Conclusions

The proposed e-voting scheme respects numerous requirements of a fair election. As a token cannot be linked with its' voter and her vote, and signing authorities are unable to link between a blinded signed vote and its' corresponding unblinded signed vote; the scheme is completely untraceable. Also, 2 different forms of signatures on a blinded token enable a voter to appear to authorities anonymously. Moreover, 2 different forms of signatures on same blinded vote prove the fairness of authorities. Even after unblinding if the vote within 2 signed forms is found meaningless, it ensures that the vote is meaningless from the beginning and intentionally submitted by the voter herself. In addition, the proposed scheme attains almost all essential requirements of e-voting in a simple way. It demonstrates that the computation time requirement for the proposed scheme is substantially small and makes the scheme scalable. A future plan of improvement is to evaluate the proposed scheme in more realistic environments where multiple authorities are distributed over different places, and many voters are involved.

References

[1] R. Abdelkader and M. Youssef, "Uvote: A ubiquitous e-voting system," in *3rd FTRA International Conference on Mobile, Ubiquitous, and Intelligent Computing (MUSIC'12)*, pp. 72–77, 2012.

[2] B. Adida, "Helios: Web-based open-audit voting," in *Proceedings of 17th USENIX Security Symposium*, Aug. 2008.

[3] K. Md. R. Alam, S. Tamura, S. Taniguchi, T. Yanase, "An anonymous voting scheme based on confirmation numbers," *IEEJ Transactions on Electronics, Information and Systems*, vol. 130, no. 11, pp. 2065–2073, 2010.

[4] R. Araujo, A. Barki, S. Brunet, and J. Traore, "Remote electronic voting can be efficient, verifiable and coercion-resistant," in *International Conference on Financial Cryptography and Data Security*, pp. 224–232, 2016.

[5] C. Burton, C. Culnane, and S. Schneider, "vvote: Verifiable electronic voting in practice," *IEEE Security Privacy*, vol. 14, pp. 64–73, July 2016.

[6] O. Cetinkaya and M. L. Loc, "Practical aspects of dynavote e-voting protocol," *Electronic Journal of E-government*, vol. 7, no. 4, pp. 327–338, 2009.

[7] D. Chaum, "Blind signatures system," *Advances in Cryptology (CRYPTO'83)*, pp. 153–156, 1983.

[8] D. L. Chaum, "Untraceable electronic mail, return addresses, and digital pseudonyms," *Communications of ACM*, vol. 24, pp. 84–90, Feb. 1981.

[9] M. R. Clarkson, S. Chong, and A. C. Myers, "Civitas: Toward a secure voting system," in *Proceedings of the 2008 IEEE Symposium on Security and Privacy*, pp. 354–368, 2008.

[10] A. Essex, J. Clark, and U. Hengartner, "Cobra: Toward concurrent ballot authorization for internet voting," in *International Conference on Electronic Voting Technology/Workshop on Trustworthy Elections (EVT/WOTE'12)*, 2012.

[11] A. Fujioka, T. Okamoto, and K. Ohta, "A practical secret voting scheme for large scale elections," in *Advances in Cryptology (AUSCRYPT'92)*, pp. 244–251, 1993.

[12] T. Granlund, *GNU Multiple Precision Arithmetic Library (GMP)*, Accessed, 2016.

[13] L. Han, Q. Xie, and W. Liu, "An improved biometric based authentication scheme with user anonymity using elliptic curve cryptosystem," *International Journal of Network Security*, vol. 19, no. 3, pp. 469–478, 2017.

[14] L. Huian, A. R. Kankanala, and X. Zou, "A taxonomy and comparison of remote voting schemes," in *23rd International Conference on Computer Communication and Networks (ICCCN'14)*, pp. 1–8, 2014.

[15] N. Islam, A. K. Md. Rokibul, and A. Rahman, "The effectiveness of mixnets-an empirical study," *Transaction on Computer Fraud and Security*, vol. 2013, no. 12, pp. 9–14, 2013.

[16] N. Islam, A. K. Md. Rokibul, and S. S. Rahman, "Commutative re-encryption techniques: Significance and analysis," *Information Security Journal: A Global Perspective*, vol. 24, no. 4, pp. 185–193, 2015.

[17] I. Jabbar and S. N. Alsaad, "Design and implementation of secure remote e-voting system using homomorphic encryption," *International Journal of Network Security*, vol. 19, no. 5, pp. 694–703, 2017.

[18] A. Juels, D. Catalano, and M. Jacobsson, "Coercion-resistant electronic elections," *Towards Trustworthy Elections*, pp. 37–63, 2010.

[19] C. Guo W. Hu L. Yuan, M. Li and Z. Wang, "A verifiable e-voting scheme with secret sharing," *International Journal of Network Security*, vol. 19, no. 2, pp. 260–271, 2017.

[20] B. Lee, C. Boyd, Ed Dawson, K. Kim, J. Yang, and S. Yoo, "Providing receipt-freeness in mixnet-based voting protocols," in *Proceedings of the Information Security and Cryptology (ICISC'03)*, pp. 245–258, 2004.

[21] C. C. Lee, T. Y. Chen, S. C. Lin, and M. S. Hwang, "A new proxy electronic voting scheme based on proxy signatures," *Lecture Notes in Electrical Engineering*, vol. 164, pp. 3–12, 2012.

[22] C. C. Lee, M. S. Hwang, and W. P. Yang, "Untraceable blind signature schemes based on discrete logarithm problem," *Fundamenta Informaticae*, vol. 55, no. 3-4, pp. 307–320, 2003.

[23] C. C. Lee, M. S. Hwang, and W. P. Yang, "A new blind signature based on the discrete logarithm problem for untraceability," *Applied Mathematics and Computation*, vol. 164, no. 3, pp. 837–841, 2005.

[24] C. T. Li and M. S. Hwang, "A secure and anonymous electronic voting scheme based on key exchange protocol," *International Journal of Security and Its Applications*, vol. 7, no. 1, pp. 59–70, 2013.

[25] C. C. Lee M. S. Hwang and Y. C. Lai, "An untraceable blind signature scheme," *IEICE Transaction on Fundamentals*, vol. E86-A, no. 7, pp. 1902–1906, 2003.

[26] J. Chen M. Wu and R. Wang, "An enhanced anonymous password-based authenticated key agreement scheme with formal proof," *International Journal of Network Security*, vol. 19, no. 5, pp. 785–793, 2017.

[27] B. Riva and A. Ta-Shma, "Bare-handed electronic voting with pre-processing," *Proceedings of the USENIX/Accurate Electronic Voting Technology Workshop*, pp. 15, 2007.

[28] A. K. Md. Rokibul and S. Tamura, "Electronic voting: Scopes and limitations," in *Proceedings of International Conference on Informatics, Electronics & Vision (ICIEV12)*, pp. 525–529, May 2012.

[29] H. A. Haddad. N. Islam S. Tamura and A. K. Md. Rokibul, "An incoercible e-voting scheme based on revised simplified verifiable re-encryption mix-nets," *Information Security and Computer Fraud*, vol. 3, no. 2, pp. 32–38, 2015.

[30] D. Sandler, K. Derr, and D. S. Wallach, "Votebox: a tamper-evident verifiable electronic voting system," in *Proceedings of the 17th USENIX Security symposium*, pp. 349–364, 2008.

[31] B. Schneier, *Applied Cryptography*, John Wiley & Sons Inc., 2nd edition edition, 2008.

[32] Wen shenq Juang, Chin laung Lei, and Pei ling Yu, "A verifiable multi-authorities secret election allowing abstaining from voting," *Computer Journal*, vol. 45, no. 6, pp. 672–682, 2002.

[33] Shinsuke Tamura and Shuji Taniguchi, "Enhanced anonymous tag based credentials," *Information Security and Computer Fraud*, vol. 2, no. 1, pp. 10–20, 2014.

Biography

Kazi Md. Rokibul Alam is currently a professor in the Dept. of Computer Science and Engineering of Khulna University of Engineering & Technology. He received Dr. (Eng.) degree in System Design Engineering from University of Fukui, Japan, and M.Sc. and B. Sc. degrees both in Computer Science and Engineering from Bangladesh University of Engineering & Technology and Khulna University, Bangladesh in 2010, 2004 and 1999, respectively. His research interests include applied cryptography, information security and machine learning.

Adnan Maruf has received his B.Sc. Eng. degree in Computer Science & Engineering from Khulna University of Engineering & Technology in 2013. From 2013 to 2016, he worked as a Sr. Software Engineer in Samsung Research & Development Institute, Banglagesh. He is currently a PhD student at Florida International University, USA. His research interests include Computer Vision, Computational Geometry, and Computer Security.

Md. Rezaur Rahman Rakib is currently doing his MS in Computer Science at Technical University of Munich (TUM) in Germany. He received his bachelor degree in Computer Science and Engineering from Khulna University of Engineering & Technology, Bangladesh in 2013. In 2014, he joined in Samsung R&D Institute Bangladesh Ltd. (SRBD) as a software engineer. In 2016, he was promoted to senior software engineer in SRBD. His research interests include neural networks and artificial intelligence, machine learning, cognitive system, and deep learning.

G. G. Md. Nawaz Ali is currently a postdoctoral research fellow with the School of Electrical and Electronic Engineering, Nanyang Technological University (NTU), Singapore. He received his PhD in the Department of Computer Science, City University of Hong Kong in 2013. He received his B.Sc. degree in Computer Science and Engineering from Khulna University of Engineering & Technology, Bangladesh in 2006. He is a member of IEEE and IEEE VTS. His current research interests

include wireless broadcasting, mobile computing, network coding, and ad hoc networking with a focus on vehicular ad hoc networking.

Peter H. J. Chong is currently the Professor and Head of the Department of Electrical and Electronic Engineering, Auckland University of Technology, New Zealand. He received the B.Eng. (with distinction) in electrical engineering from the Technical University of Nova Scotia, Halifax, NS, Canada, in 1993, and the M.A.Sc. and Ph.D. degrees in electrical engineering from the University of British Columbia, Vancouver, BC, Canada, in 1996 and 2000, respectively. Between 2000 and 2001, he worked at Agilent Technologies Canada Inc., Canada. From 2001 to 2002, he was at Nokia Research Center, Helsinki, Finland. From May 2002 to 2016, he was with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore as an Associate Professor (Tenured). He was an Assistant Head of Division of Communication Engineering between 2011 and 2013, since July 2013 to April 2016, he was the Director of Infinitus, Centre for Infocomm Technology in School of EEE. He has visited Tohoku University, Japan, as a Visiting Scientist in 2010 and Chinese University of Hong Kong (CUHK), Hong Kong, between 2011 and 2012. He is currently an Adjunct Professor of CUHK.

Yasuhiko Morimoto is a professor at Hiroshima University. He received his B.E., M.E. and Ph.D degrees from Hiroshima University in 1989, 1991 and 2002 respectively. From 1991 to 2002, he had been with IBM Tokyo Research Laboratory where he worked for data mining project and multimedia database project. Since 2002, he has been with Hiroshima University. His current research interest includes data mining, machine learning, geographic information system and privacy preserving information retrieval.

Multiple New Formulas for Cipher Performance Computing

Youssef Harmouch[1], Rachid Elkouch[1], Hussain Ben-azza[2]
(Corresponding author: Youssef Harmouch)
Department of Mathematics, Computing and Networks, National Institute of Posts and Telecommunications[1]
10100, Allal El Fassi Avenue, Rabat, Morocco
(Email:harmouch@inpt.ac.ma)
Department of Industrial and Production Engineering, Moulay Ismail University[2]
National High School of Arts and Trades, Mekns, Morocco
(Received Apr. 03, 2017; revised and accepted July 17, 2017)

Abstract

Cryptography is a science that focuses on changing the readable information to unrecognizable and useless data to any unauthorized person. This solution presents the main core of network security, therefore the risk analysis for using a cipher turn out to be an obligation. Until now, the only platform for providing each cipher resistance is the cryptanalysis study. This cryptanalysis can make it hard to compare ciphers because each one is vulnerable to a different kind of attack that is often very different from others. Our contribution in this paper is to develop new risk analysis formulas to offer a theoretical background for both the cipher designer and the simple users. Those formulas will help to suggest a fair platform for measuring risk, safety, complexity and cost, in order to determine a quantifiable value for performance to each cipher. This can lead to a fair comparison in a fair scale.

Keywords: Complexity & Cost; Quantifiable-Value; Risk Analysis; Security-Level; Security Performance

1 Introduction

Although, humans today constantly depend on computer technology in their life, they continue to have a hard follow to security aspects between different technologies. This is caused by the tiny ability to compare, to contrast, and to make quantifiable statements about security systems. This means that having a fixed global model for information security is extremely valuable for having a basis to determine where to put limited resources, pay attention, and how to best secure systems.

However, risk analysis (quantitative or qualitative [16, 19]) remains a difficult problem, since computer security is a multidimensional attribute (confidentiality, availability, integrity, non-rejection, accountability, authenticity, reliability of IT systems, *etc.*). Moreover, these dimensions are not necessarily commensurate properties. For example, an online newspaper will be primarily interested in the integrity of their information while a financial stock exchange network may define their security as real-time availability and information privacy [14, 23]. This means that, the many facets of the attribute must all be identified and adequately addressed. Furthermore, the security attributes are terms of qualities, thus measuring such quality terms need a unique identification for their interpretations meaning [20, 24]. Besides, the attributes can be interdependent. The first thing is to identify a set of security-related attributes that are important to the use of the system. This leads to decide whether the system security must be represented as a vector or as a single value.

In large systems, risk analysis becomes a very painful task. The remaining solution to use the decomposition method to develop simple, small and stand alone components of the system. Therefore, in order to better measure risk analysis of the systems, it is necessary to seek for the common ground between all the systems and their components. This common ground is the security protocols or algorithms. Since this latter is based on computer, mathematical and/or logical operations, the scale for risk analysis should be changed from **macro scale** (network application, software, threats, hardware, protocols, *etc.*) to **micro scale** (cipher and algorithm).

Thus, this research studies the cipher risk analysis upon **MLO** (**M**athematical or/and Computer **L**ogical **O**peration). This paper proposes new risk formulas that represent an improvement in security quantification. By calculating the security level offered by each MLO, the cryptograph can easily choose which MLO to use and where to place it, so as to increase the complication of the cipher/algorithm within the developing phase before applying any cryptanalysis studies. This paper provides a fair comparison of **security**, **risk** and utilizing **cost** for several ciphers based on theirs MLO while respecting each

cipher properties. These properties englobe inner structures, key space, round number, complexity, successful cryptanalysis attacks, *etc.* to provide a value that determines the **safety degree** of each cipher, in order to put a fair platforms where ciphers can be judged and compared precisely.

The paper is organized as follows. In Section 2, we introduce some concepts of cryptography, then we define the different structures utilized by ciphers followed with an introduction of the most knowing cryptanalysis attacks, after that, we introduce the ciphers used in this study. In Section 3, we investigate the study and the development of the new formulas for risk analysis, while in Section 4, we focus on results and discussion. We conclude the paper in Section 5.

2 State of Art and Motivation

Because information privacy has become a major concern for both users and companies, cryptography is considered as a standard for providing information trust, security, electronic financial transactions, controlling access to resources and stopping non-authorized persons from obtaining critical or private information. It must be mentioned that the strength of the cryptography algorithm depends on the length of the key, secrecy of the key, the complexity of the process and how they all work together [18].

Ciphers differ with their construction structure. This leads to different types of comportment. These structures can be organized as (for symmetric cryptography):

- Permutation network: is when a cipher uses a permutation box (P-box). This latter is used to permute or transpose data across plaintext, retaining diffusion while transposing [9].
- Substitution network: is when a cipher uses a substitution box (S-box). It is used to obscure the linearity between the key and the ciphertext [7, 8].
- Substitution permutation network: is when a cipher uses both S-box and P-box in its encryption function.
- Feistel Network: is when a cipher uses a Feistel scheme. It is a technique used in the construction of block cipher-based algorithms and mechanisms [13]. If the two blocks (left and right) are not of equal length, then the scheme is called unbalanced Feistel scheme.
- Lai-Massey scheme: is when a cipher uses a Lai-Massey scheme [26].

During the cipher design phase, the cryptograph applies one or more structures, to determine the security level offered by the cipher besides its behavior. Those structures in addition to nonlinear functions are an important functionality that each cipher must have in order to put confusion and diffusion alongside, to prevent the finding of any linear link between plaintext and ciphertext so as to increase the complexity of breaking the cipher.

Cryptanalysis tests the weakness of the cryptosystem by trying to break it without any knowledge of the key used. The most popular attack is the brute force where the cyber criminal tries every possible key to break ciphers; therefore, the only way to resist is by enlarging the key space to make it infeasible [21].

Thus, the question that needs to be asked is, "Which cipher is the best in security term and how can we measure its safety?" The ability to compare and/or to make quantifiable statements about system security is extremely valuable, since it offers a basis to determine how to best secure the systems. Besides, the complete understanding of a subject cannot be done with neither measurements nor quantifying value as written by Lord Kelvin in 1883: "When you can measure what you are speaking about and express it in numbers you know something about it, but when you cannot measure it, when you cannot express it in numbers, your knowledge is of a meager and unsatisfactory kind" [23].

The development of the theory of measurements and quantifying cipher is the main motivation for this research. However, it is a difficult problem due to The large number of structure and operation that cipher utilizes. Moreover, in this research, the only trust measurements used is based over probabilities.

Furthermore, this research alongside with cryptanalysis tries to rate each cipher based on inner structure, key space, successful attacks, *etc.* in order to have a fair platform for comparison and does not try to replace the cryptanalysis, which is still the only science that tests the cipher weakness.

In this work, we try to answer the following questions: if a user has two ciphers, A1 and A2 while knowing that the best successful cryptanalysis attack for A1 resp. A2 is B1 resp. B2 and the two attacks have the same successful rate, what is the best cipher to choose? If A1 is a bit quicker than A2 and B2 is a little less successful than B1, what is the most optimal cipher for a system/network? Another question, if a user has multiple ciphers A1, A2, $\cdots$, An, what is the order of the most suitable cipher to her/his own system/network? i.e. what is the order of ciphers which offers an acceptable resistance (not always the best) to cryptanalysis attacks, and which is the most suitable to the system/network (real time application, Full HD conferencing, high throughput network, data-center file encryption, *etc.*)?

Even if risk analysis proposed in this paper can be used for any type of encryption (symmetric, asymmetric, bloc, stream), this paper focuses on symmetric block ciphers like [12] "AES, Blowfish, Camellia, CAST-128/256, DES/3DES, GOST, IDEA, MARS, RC2, RC5, RC6, Serpent, SHACAL2, SHARK, SKIPJACK, Three-way, Twofish, and XTEA". Each cited cipher will be revised respecting each one's properties; such as key length, block length and the mode of operation.

3 Risk Analysis

Risk analysis is a technique used to identify and assess factors that may put at risk the safety of security based upon a cipher. This technique helps to define the optimal cipher in order to reduce the probability for these factors from occurring. Therefore, in this section we define multiple indexes factor to help studying every cipher either in the design phase or in comparison with those that already exist. These factors are called **index of safety**, **index of risk**, **complexity** and **cipher cost** and will be defined in the following paragraph.

3.1 Index of Safety (IS)

IS defines the level of security factor that a cipher offers to users, that is to say, this factor studies the robustness of the cipher structure. It consists of round number (R), key-block index (K/B) that defines the length of key per length of data block, and the structure type index (S) such as Feistel, P-box, S-box, *etc.* multiplied by their factors and the number of uses in one round. As so and before defining IS, several definitions must be provided:

Definition 1. *We define ρ as the break-probability for a structure or operation used in encryption process. i.e. ρ is equal to probability of extracting the plaintext from the ciphertext after applying a structure or an operation to the plaintext.*

The following Table 1 shows ρ for different encryption operations where "m" is the block length in bits and "ξ" defines the modulus.

Table 1: Break probability for structure type or operation

Structure type	Break Probability
AND	$1/2^{2m}$
OR	$1/2^{2m}$
XOR	$1/2^{m}$
Concatenation	1
Modular addition	$1/\xi^{\frac{m}{log_2\xi}}$
Modular subtraction	$1/\xi^{\frac{m}{log_2\xi}}$
Modular multiplication	$1/\xi^{\frac{2m}{log_2\xi}}$
Modular exponentiation	$1/\xi^{\frac{2m}{log_2\xi}}$
Left or Right rotation	$1/(m-1)$
NOT operation	1
Conditional NOT operation	$1/2^{m}$
Permutation box	$1/2^{m-1}$
Substitution box	$1/(2^m-1)$
balanced Feistel schema	$1/2^{\frac{m}{2}}$
Lai-Massey schema	$1/2^{2m}$

Since each structure presents a different bit operation and is linearly correlated to both "bit number and operation type", the structure/operation resistance can be measured through ρ.

Note 1. *The proof of the results listed in Table 1 is presented in Appendix-A.*

Definition 2. *We define the measurement of the resistance factor S for a structure or operation by:*

$$S = -log_2\rho \tag{1}$$

where ρ is the break probability (see Table 2 below).

Since the "1/2" is common in all type of structure due to binary representation, it does not provide any utility for comparison. Remark that, the only valuable information in ρ expression is "m" or "ξ". As so, in Equation (1), we use "$-log_2$" to remove the "1/2" and get the useful information (m, ξ) for the future study.

Table 2: Resistance factor for structure type or operation

Structure type	Resistance Factor
AND	$2m$
OR	$2m$
XOR	m
Concatenation	0
Modular addition	m
Modular subtraction	m
Modular multiplication	$2m$
Modular exponentiation	$2m$
Left or Right rotation	$log_2(m-1) \approx log_2 m$
NOT operation	0
Conditional NOT operation	m
Permutation box	$m-1$
Substitution box	$log_2(2^m-1) \approx m$
balanced Feistel schema	$m/2$
Lai-Massey schema	m

The resistance factor enlarge the scale from [0, 1] for ρ to [0, $+\infty$[with valuable conservation of bit information "m". therefore, the best resistance factor unity is "bit".

Definition 3. *We define the **Block-Round Function BRF** as the block contains one or more successive functions that have the same round number.*

Example 1. *Let us define A, B and C as a function or a bit operation and "r1", "r2" as the round number (r1≠r2). According to the following algorithms, we define BRF for each algorithm as showed in following Table 3.*

Definition 4. *We define the resistance factor efficiency **ES** for one BRF by:*

$$ES = \frac{1}{\lambda}\sum_{i=1}^{\lambda} S_i \tag{2}$$

where S_i is the resistance factor for the structure or operation number "i" and λ is the total number of structures or operations in one BRF block.

Using Equation (1) in Equation (2) gives

Table 3: BRF definition example

Description	Algorithm.1	Algorithm.2	Algorithm.3	Algorithm.4
Algorithm Body	for i from 0 to r1 repeat {A;B} end repeat C	for i from 0 to r1 repeat {A;B;C} end repeat	for i from 0 to r1 repeat {A;B} end repeat for i from 0 to r2 repeat {C} end repeat	for i from 0 to r1 repeat {A;B;C} end repeat for i from 0 to r2 repeat {A;B;C} end repeat
Number of BRF	2	1	2	2
BRF function	BRF1={A, B} BRF2={C}	BRF1={A, B, C}	BRF1={A, B} BRF2={C}	BRF1={A, B, C} BRF2={A, B, C}

$$ES = \frac{1}{\lambda}\sum_{i=1}^{\lambda} S_i = -\frac{1}{\lambda}\sum_{i=1}^{\lambda} log_2\rho_i$$

where ρ_i is the break probability for the structure or operation number "i".

Note 2. *ES indicates the efficacy for a BRF .i.e. ES indicates the level of security at which a function or operation affects the rest of the functions in a BRF, and so, ES helps to measure the security offered through a BRF. ES and S have the same unit "bits".*

Definition 5. *To determines the potential security-level offered by a cipher, we define the key-block factor **KB** for a cipher by:*

$$KB = \alpha/\beta \tag{3}$$

where α is the bit number of the key and β is the bit number of the plaintext block.

Note 3. *Equation (3) do not consider the number or the length of the sub-keys. According to that, and as an example AES-128 has KB=1, AES-192 has KB=1.5 and AES-256 has KB=2.*

Definition 6. *We define the total resistance factor **TS** for a BRF number "i" by:*

$$TS_i = R_i \times ES_i \tag{4}$$

where R_i is the round number for the BRF number "i".

Note 4. *Since ES unity is "bit" and R is just a number, we propose a new unity for TS called RB "Round Bit".*

Generally, a cipher is a composition of one or more BRFs. Hence, the cipher total resistance factor is obtained by adding all its BRFs total resistance factor. Thus, Equation (4) gives:

$$TS = \sum_{i=1}^{\mu} TS_i = \sum_{i=1}^{\mu} r_i \times ES_i = \sum_{i=1}^{\mu} r_i \times \frac{1}{\lambda_i}\sum_{i=1}^{\lambda_i} S_i \tag{5}$$
$$= -\sum_{i=1}^{\mu} r_i \times \frac{1}{\lambda_i}\sum_{i=1}^{\lambda_i} log_2\rho_i$$

where "μ" is the BRF number in the cipher encryption process, "λ_i" is the number of structures or operations for the BRF number "i" and "r_i" is the round number for the BRF number "i".

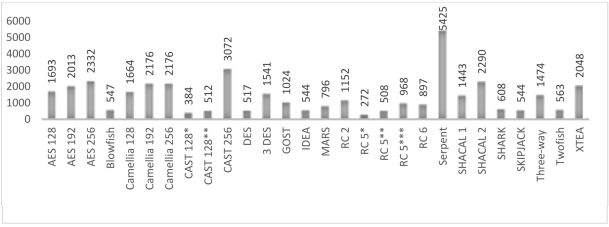


Figure 1: TS measurement for several ciphers

Figure 1 illustrates the TS calculated for all the studied ciphers, while respecting the plaintext block length of each cipher. Figure 1 uses the equations taken from the following Table 4. The equations in Table 4 are calculated using Equation (5).

Figure 1 shows that the serpent presents the highest TS factor followed by CAST-256 and AES-256, this result means that serpent uses more complicated and complex inner structure and/or more round numbers, which yields to more ciphertext-complexity and thus increasing the resistance-probability to cryptanalysis attacks. Note that, it is not always the highest TS that is the more secure, because TS does not provide any information about successful cryptanalysis attacks applied over the cipher and its success rate.

Note 5. *In Figure 1, we use the terminology:*

- *CAST-128*/CAST-128** is for key length from 40 to 80/80 to 128 bits.*
- *RC5*/RC5**/RC5*** is for plaintext length 32/64/128 bits.*

Definition 7. *We define the index of security for a cipher by:*

$$IS = log_{10}(KB \times TS) \tag{6}$$

Note 6. *The goal of the logarithm scale used in Equation (6) is to reduce the vast values obtained from computed IS. Moreover we define a new unity of IS called **SC** "**S**ecurity per **C**ipher".*

Accordingly, Table 5 englobes IS measurement for several ciphers which can be observed graphically in the following Figure 2.

Table 4: Total resistance factor for several ciphers

Cipher	TS (n: Length of the plaintext block in bits)
AES 128	$(129n-31)/12$
AES 192	$(189n-37)/12$
AES 256	$(219n-43)/12$
Blowfish	$(17n+log_2(n)-1)/2$
Camellia 128	$13n$
Camellia 192	$17n$
Camellia 256	$17n$
CAST 128	$6n$ if 40bits $\leq$ keysize $<$ 80bits $8n$ if 80bits $\leq$ keysize $<$ 128bits
CAST 256	$24n$
DES	$8n+log_2(n)-1$
3DES	$24n+log_2(n)-1$
GOST	$16n$
IDEA	$\frac{17}{2}n$
MARS	$(41n+log_2(n)+56log_2(13)+14)/7$
RC 2	$18n$
RC 5	$n+[1\rightarrow 255](log_2(n)-1+n/2)$
RC 6	$(47n+40log_2(n)-20)/7$
Serpent	$\frac{1}{3}(127n+1+log_2(3)+log_2(5)+2log_2(7)+$ $log_2(11)+log_2(13))$
SHACAL 1	$(35n+40log_2(n)-120)/4$
SHACAL 2	$(35n+40log_2(n)-120)/4$
SHARK	$19n/2$
SKIPJACK	$8(n+log_2(n)-2)$
Three-way	$(77n-11)/5$
Twofish	$(25n+32log_2(n-4)+3log_2(n)-67)/6$
XTEA	$32n$

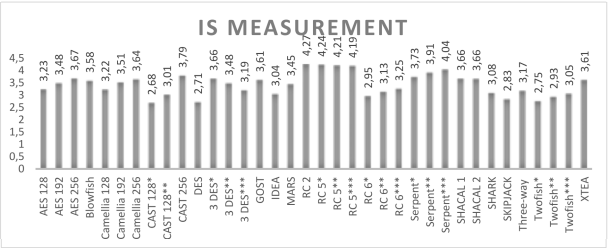


Figure 2: IS measurement for several ciphers

Just like TS, IS provides a scale to measure the possibility of being more secure to cryptanalysis attacks by collecting different information (TS, key). This collection focuses on the complexity of the cipher body and the key space without any examination to cryptanalysis attack neither to the nature of the cipher body. This is why in Figure 2, RC2 and RC5 shows more IS value than Serpent, and 3DES shows more IS value than AES-192.

It is clearly observed that IS does not provide all information for best cipher rating, that is why another factor is needed. This factor will measure the risk of using the cipher by studying its best-known successful cryptanalysis attacks. This measure is explained in the following section under the name of **I**ndex of **R**isk (**IR**).

3.2 Index of Risk (IR)

As each cipher has a different structure, many different cryptanalysis attacks are invented and developed including: "Linear cryptanalysis [17], Differential cryptanalysis [1], Differential-linear cryptanalysis, Impossible differential cryptanalysis, Truncated differential cryptanalysis [5], Integral cryptanalysis, Higher-order differential cryptanalysis [25], Meet-in-the-middle [2], Slide attack [4], Boomerang Attack [22], Related Key Attack [3], Mod n [15], XSL [6], Frequency analysis [11], The index of coincidence, Chi-square test [10], *etc.*"

The major differences between those attacks above make it difficult to fairly judge and compare cipher. As so, in order to create a credible scale, we define a new term called the Index of Risk IR.

IR defines the measure of the risk of using a cipher. It combines the success rates of the most successful cryptanalysis attacks and the security index that the ciphers offers. However before defining IR, several definitions must be mentioned.

Definition 8. *We define* ***BA*** *as the best success rate factor for a multiple cryptanalysis attacks by:*

$$BA = 1 - \frac{\min_{i\in[0,\tau-1]} log_2(CCA_i)}{key\ lenght\ in\ bits} \tag{7}$$

where τ presents the number of cryptanalysis attacks while CCA_i is the computational complexity of the attack number "i".

CCA_i divided by the key-length in Equation (7) presents the success rate factor or the percentage rate for a successful cryptanalysis attack. To show this, we compute BA based on data taken from Table 2 in the paper [12]. Figure 3 contains the computing result:

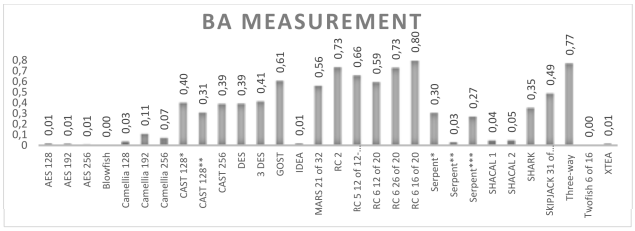


Figure 3: BA measurement for several ciphers

Definition 9. *We define the index of risk (IR) for a cipher as:*

$$IR = 100 \times \frac{BA}{IS} \tag{8}$$

Equation (8) takes into consideration two factors: the success rate for cryptanalysis attacks and the measured index of security across the body of the cipher. The number 100 is just a coefficient to enlarge the scale since dividing the rate for a successful cryptanalysis attack by the safety of the cipher body gives results always less than "1". Figure 4 shows the calculated IR.

Table 5: IS calculating for several ciphers per key length and plaintext block length

Cipher	Key Length (bits)	Block Length (bits)	TS (RB)	KB	IS for multiple case (SC)	max IS (SC)
AES 128	128	128	1693, 42	1	3, 22876383	3, 22876383
AES 192	192	128	2012, 92	1, 5	3, 479917055	3, 47991705
AES 256	256	128	2332, 42	2	3, 668836132	3, 66883613
Blowfish	32 → 447	64	546, 5	0, 5 → 6, 98	(2, 43656, 3, 5817)	3, 58171772
Camellia 128	128	128	1664	1	3, 221153322	3, 22115332
Camellia 192	192	128	2176	1, 5	3, 51375015	3, 51375015
Camellia 256	256	128	2176	2	3, 638688887	3, 63868889
CAST 128*	40 → 80	64	384	0, 63 → 1, 25	(2, 38021, 2, 6812)	2, 68124124
CAST 128**	80 → 128	64	512	1, 25 → 2	(2, 80618, 3, 0103)	3, 01029996
CAST 256	138 → 256	128	3072	1, 08 → 2	(3, 52009, 3, 7885)	3, 78845121
DES	64	64	517	1	2, 713490543	2, 71349054
3 DES*	192	64	1541	3	3, 664923893	3, 66492389
3 DES**	124	64	1541	1, 9375	3, 47504435	3, 47504435
3 DES***	64	64	1541	1	3, 187802639	3, 18780264
GOST	256	64	1024	4	3, 612359948	3, 61235995
IDEA	128	64	544	2	3, 036628895	3, 0366289
MARS	128 → 448	128	796 → 318	1 → 3, 5	2, 90109 → 3, 4452	3, 44515447
RC 2	8 → 1024	64	1152	0, 13 → 16	2, 15836 → 4, 2656	4, 26557246
RC 5*	8 → 2040	32	272	0, 25 → 63, 8	1, 83251 → 4, 239	4, 23904909
RC 5**	8 → 2040	64	508	0, 13 → 31, 9	1, 80277 → 4, 2093	4, 20931391
RC 5***	8 → 2040	128	968	0, 06 → 15, 9	1, 78176 → 4, 1883	4, 18829556
RC 6	128	128	896, 571	1	2, 952584895	2, 95258489
RC 6	192	128	896, 571	1, 5	3, 128676154	3, 12867615
RC 6	256	128	896, 571	2	3, 253614891	3, 25361489
Serpent*	128	128	5425, 09	1	3, 734406852	3, 73440685
Serpent**	192	128	5425, 09	1, 5	3, 910498111	3, 91049811
Serpent***	256	128	5425, 09	2	4, 035436848	4, 03543685
SHACAL 1	128 → 512	160	1443, 22	0, 8 → 3, 2	3, 06242 → 3, 6645	3, 6644823
SHACAL 2	128 → 512	256	2290	0, 5 → 2	3, 05881 → 3, 6609	3, 66086548
SHARK	128	64	608	2	3, 084933575	3, 08493357
SKIPJACK	80	64	544	1, 25	2, 832508913	2, 83250891
Three-way	96	96	1474	1	3, 168497484	3, 16849748
Twofish*	128	128	562, 756	1	2, 750319913	2, 75031991
Twofish**	192	128	562, 756	1, 5	2, 926411172	2, 92641117
Twofish***	256	128	562, 756	2	3, 051349909	3, 05134991
XTEA	128	64	2048	2	3, 612359948	3, 61235995

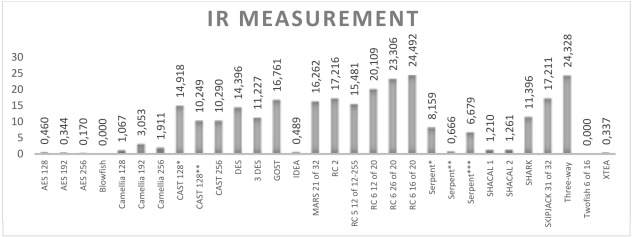


Figure 4: IR measurement for several ciphers

3.3 Cipher Cost (CC)

CC defines the cost of using a cipher for a system/network. It depends on IR and complexity. The most developed cipher is normally working only on a fixed block size of plaintext. This takes approximately the same time for encryption/decryption independently of input (ECB mode), thus they are O(1).

Even if we put them into a mode of operation to encrypt a longer plaintext, we usually get an O(m) complexity, where "m" is the plaintext size, as we have O(m) blocks of data to encrypt. This O(m) presents the minimum, because each cipher has to encrypt at least each input-bit once, to be reversible, even if different modes of operations have different complexity (Triple-DES usually needs three times the computing power as DES, but still then O(1) or O(m)).

As a result, the uses of O becomes useless to compare the complexity between ciphers. Thus, from now on, we redefine the **complexity** as "**the number of CPU cycle needed to encrypt the plaintext**". Since the plaintext size differs from cipher to another, we set the plaintext size required for the complexity measurement as one Mega-Byte. Besides, this complexity is linearly linked to the computing power or computing time, as so it allows multiple usages for it.

Furthermore, the complexity must have a **reference point** in order to allow a future comparison. This reference point will be the complexity of encrypting the plaintext with XOR operation, since XOR is the fastest strong easy low-power consumption and simple cryptographic computer operation. Thus, we define the normalized complexity Γ as the ratio between the complexity of the cipher and the complexity of XOR:

$$\Gamma = \frac{Cipher\ complexity}{XOR\ complexity} \tag{9}$$

Since this paper is interested in putting a quantifiable value to cipher performance, the Γ measurement for ciphers (studied in this paper) from Equation (9) must be standardized (standard score), because we are only interested in choosing the less cipher-complexity compared to others. Thus, Γ becomes $\underline{\Gamma}$:

$$\underline{\Gamma} = \Gamma/\sigma \tag{10}$$

where σ is the standard deviation of Γ for all studied ciphers.

Note 7. *There is no need for the subtraction of* Γ *by the mean "*μ*" in Equation (10) because it presents just a shift scale by "*$-\mu/\sigma$*".*

Table 6 shows the measurement of $\underline{\Gamma}$ for all studied ciphers, this measurement is illustrated in Figure 5.

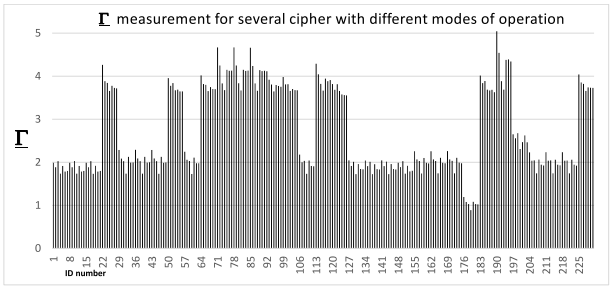


Figure 5: $\underline{\Gamma}$ measurement for several ciphers with different modes of operation

The experimental environment for the complexity measurement was a C++ code application developed in Microsoft Visual studio 2010 for Windows 7 desktop and GCC 4.8.2 for Centos7 for Linux OS. The ciphers used in this study are taken from two version of an open source library called Crypto++ (cryptopp5.6.2 and cryptopp5.6.3). The test was running under different machine (from Intel core2 until Intel core i5). As observed from our experimental results, the changing of OS affects the complexity in about 10%, while the changing of library affects less than 4%. The most important change in complexity was when changing the tested machines (up to 70%). $\underline{\Gamma}$ decreases the difference between results in less than $0,3\%$. This tiny difference makes the values in Table 6 a trustful result to calculate the cost of using every studied ciphers.

Definition 10. *We define the cipher cost (**CC**) by:*

$$CC = IR \times \underline{\Gamma} \tag{11}$$

4 Results & Discussion

The CC study in Equation (11) englobes the recognition of many parameters (safety, speed, resistance, risk...) for ciphers in different modes of operation. This helps to provide a good platform to compare ciphers with many considered variables as sizes of data blocks, key size, type of cipher, complexity, round number, successful cryptanalysis attacks...

CC, IS, complexity ..., and IR present a theoretical and logical MLO formulas for studying risk analysis. These formulas will help to obtain quantifiable values, so as to support either a cipher designer or a normal user to choose the most optimal ciphers to his/her network/system. Since each parameter has a different definition and interpretation, we define each parameter unity as following (see Table 7).

Figure 6 illustrates the data presented in Table 8. It shows the CC values for all studied ciphers with their different operation modes.

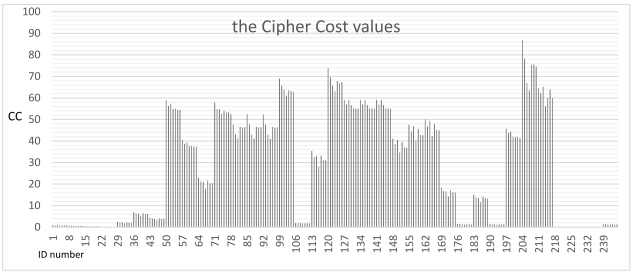


Figure 6: CC measurement for several ciphers with differents mode of operation

Figure 6 shows that CC is linearly related to the applied mode of operation (CBC-CTS, CBC, CFB-FIPS, CFB, CTR, ECB and OFB) and the used cipher. For example, Camellia, MARS, RC2 and SKIPJACK show less cost in FIPS than CTR as opposed to AES, DES, GOST, IDEA, RC5/6, Three-Way and XTEA that show more cost in FIPS than CTR. In addition, we notice that Twofish with 128 bits in key has less complexity than Twofish with 192/256 bits in key and the three have the same cost. This is due to the lack of a successful cryptanalysis attack which also makes for instance both

Table 6: Γ measurement for several cipher with different mode of operation

ID	Cipher/mode	Complexity	Γ	$\bar{\Gamma}$	ID	Cipher/mode	Complexity	Γ	$\bar{\Gamma}$
1	AES 128/CBC-CTS	606618769	239,112846	1,98722827	117	RC2/CTR	1203338412	474,323722	3,9420279
2	AES 128/CBC	575240588	226,744409	1,88443619	118	RC2/ECB	1186101033	467,529209	3,88555981
3	AES 128/CFB-FIPS	618371813	243,745581	2,02573018	119	RC2/OFB	1193908739	470,606798	3,91113715
4	AES 128/CFB	529738052	208,80853	1,73537399	120	RC5/CBC-CTS	1164665677	459,079967	3,81533952
5	AES 128/CTR	584754793	230,494653	1,91560386	121	RC5/CBC	1123856089	442,993922	3,681651
6	AES 128/ECB	544337846	214,563377	1,7832016	122	RC5/CFB-FIPS	1163470732	458,608952	3,81142499
7	AES 128/OFB	549277793	216,510571	1,79938442	123	RC5/CFB	1116663426	440,158767	3,65808848
8	AES 192/CBC-CTS	606807867	239,187383	1,98784774	124	RC5/CTR	1090960903	430,027522	3,57388934
9	AES 192/CBC	575165113	226,714658	1,88418894	125	RC5/ECB	1086344652	428,20792	3,55876692
10	AES 192/CFB-FIPS	619090484	244,028861	2,02808448	126	RC5/OFB	1085131939	427,729901	3,55479418
11	AES 192/CFB	529362463	208,660482	1,7341436	127	RC6 128/CBC-CTS	623916055	245,930972	2,04389262
12	AES 192/CTR	584423615	230,364112	1,91451895	128	RC6 128/CBC	583720958	230,087143	1,91221711
13	AES 192/ECB	544565674	214,65318	1,78394794	129	RC6 128/CFB-FIPS	614196450	242,099764	2,01205207
14	AES 192/OFB	549168145	216,467351	1,79002522	130	RC6 128/CFB	526859440	207,673858	1,72594392
15	AES 256/CBC-CTS	606496990	239,064844	1,98682934	131	RC6 128/CTR	597527297	235,529231	1,95744543
16	AES 256/CBC	574979302	226,641417	1,88358024	132	RC6 128/ECB	562795685	221,838962	1,84366781
17	AES 256/CFB-FIPS	618415537	243,762815	2,02587342	133	RC6 128/OFB	560116938	220,783072	1,83489248
18	AES 256/CFB	528806401	208,441298	1,73232199	134	RC6 192/CBC-CTS	623060290	245,593652	2,04108921
19	AES 256/CTR	585007046	230,594084	1,91643022	135	RC6 192/CBC	582755660	229,706648	1,90905488
20	AES 256/ECB	544449343	214,607326	1,78356685	136	RC6 192/CFB-FIPS	613773707	241,93313	2,0106672
21	AES 256/OFB	548895796	216,359998	1,79813303	137	RC6 192/CFB	526272042	207,442321	1,72401966
22	Blowfish/CBC-CTS	1301529523	513,028024	4,26369311	138	RC6 192/CTR	597128229	235,37193	1,95613812
23	Blowfish/CBC	1185771135	467,399172	3,88447909	139	RC6 192/ECB	561763610	221,432145	1,84028683
24	Blowfish/CFB-FIPS	1172637409	462,22221	3,84145419	140	RC6 192/OFB	559019148	220,350352	1,83129622
25	Blowfish/CFB	1116176795	439,966951	3,65649432	141	RC6 256/CBC-CTS	623773662	245,874844	2,04342615
26	Blowfish/CTR	1152278879	454,197424	3,77476149	142	RC6 256/CBC	583168230	229,869272	1,91040643
27	Blowfish/ECB	1138379002	448,718465	3,72922675	143	RC6 256/CFB-FIPS	614791834	242,334449	2,0140025
28	Blowfish/OFB	1134352213	447,131213	3,71603535	144	RC6 256/CFB	526860794	207,674392	1,72594835
29	Camellia 128/CBC-CTS	697684950	275,008692	2,28555285	145	RC6 256/CTR	597539354	235,533984	1,95748493
30	Camellia 128/CBC	636038910	250,709476	2,083606	146	RC6 256/ECB	562755641	221,823177	1,84353663
31	Camellia 128/CFB-FIPS	618211948	243,682566	2,02520648	147	RC6 256/OFB	559793120	220,655431	1,83383168
32	Camellia 128/CFB	529630608	208,766178	1,73502201	148	Rijndael/CBC-CTS	607193558	239,339412	1,98911123
33	Camellia 128/CTR	648897561	255,778012	2,12572978	149	Rijndael/CBC	575906086	227,00673	1,8866163
34	Camellia 128/ECB	607755943	239,561089	1,99095355	150	Rijndael/CFB-FIPS	618801502	243,914952	2,02713781
35	Camellia 128/OFB	608991847	240,04825	1,99500226	151	Rijndael/CFB	529918472	208,879646	1,73596503
36	Camellia 192/CBC-CTS	699138767	275,581747	2,29031543	152	Rijndael/CTR	585698760	230,86674	1,91869621
37	Camellia 192/CBC	637770278	251,391935	2,0892778	153	Rijndael/ECB	545166760	214,890112	1,78591704
38	Camellia 192/CFB-FIPS	618128071	243,649504	2,02493171	154	Rijndael/OFB	549827906	216,727411	1,80118654
39	Camellia 192/CFB	529521260	208,723076	1,7346638	155	Serpent 128/CBC-CTS	688479604	271,380191	2,25539697
40	Camellia 192/CTR	649489083	256,011174	2,12766755	156	Serpent 128/CBC	630447707	248,505574	2,06528972
41	Camellia 192/ECB	607876365	239,608556	1,99134804	157	Serpent 128/CFB-FIPS	619105029	244,034595	2,02813213
42	Camellia 192/OFB	610489632	240,638636	1,99990887	158	Serpent 128/CFB	530848795	209,246355	1,73901268
43	Camellia 256/CBC-CTS	697840331	275,069939	2,28606187	159	Serpent 128/CTR	640757436	252,569393	2,09906346
44	Camellia 256/CBC	637244769	251,184794	2,08755628	160	Serpent 128/ECB	606671840	239,133765	1,98740213
45	Camellia 256/CFB-FIPS	618300350	243,717412	2,02549608	161	Serpent 128/OFB	602917669	237,653972	1,9751038
46	Camellia 256/CFB	528796556	208,437417	1,73228974	162	Serpent 192/CBC-CTS	688656357	271,449862	2,255976
47	Camellia 256/CTR	648907402	255,781891	2,12576201	163	Serpent 192/CBC	630382163	248,479738	2,065075
48	Camellia 256/ECB	606957069	239,246195	1,98833651	164	Serpent 192/CFB-FIPS	619989498	244,383229	2,03102957
49	Camellia 256/OFB	609154798	240,11248	1,99553607	165	Serpent 192/CFB	531868536	209,648309	1,74235326
50	CAST128/CBC-CTS	1207115268	475,812457	3,95440054	166	Serpent 192/CTR	641784145	252,974094	2,10242687
51	CAST128/CBC	1152874571	454,432229	3,77671292	167	Serpent 192/ECB	606879786	239,215732	1,98808334
52	CAST128/CFB-FIPS	1171466078	461,760503	3,83761702	168	Serpent 192/OFB	603391023	237,840555	1,97665447
53	CAST128/CFB	1120645041	441,728213	3,67113189	169	Serpent 256/CBC-CTS	689815936	271,906937	2,25977467
54	CAST128/CTR	1125292321	443,560046	3,68635596	170	Serpent 256/CBC	631092386	248,759689	2,06740163
55	CAST128/ECB	1113342081	438,849582	3,64720806	171	Serpent 256/CFB-FIPS	620027224	244,398099	2,03115316
56	CAST128/OFB	1113495774	438,910164	3,64771154	172	Serpent 256/CFB	532203752	209,780442	1,7434514
57	CAST256/CBC-CTS	685091968	270,044876	2,24429938	173	Serpent 256/CTR	641843403	252,997452	2,10262099
58	CAST256/CBC	628014462	247,546454	2,05731863	174	Serpent 256/ECB	607894806	239,615825	1,99140846
59	CAST256/CFB-FIPS	619465699	244,176761	2,02931365	175	Serpent 256/OFB	604023419	238,089829	1,97872614
60	CAST256/CFB	528010235	208,12747	1,72971382	176	SHACAL2/CBC-CTS	363831931	143,412787	1,19188053
61	CAST256/CTR	643415330	253,617063	2,10777048	177	SHACAL2/CBC	329080009	129,714512	1,07803637
62	CAST256/ECB	603021952	237,695077	1,97544543	178	SHACAL2/CFB-FIPS	314109897	123,813695	1,02899564
63	CAST256/OFB	603737641	237,977182	1,97778996	179	SHACAL2/CFB	271423257	106,987767	0,88915806
64	DES/CBC-CTS	1226965614	483,636931	4,01942848	180	SHACAL2/CTR	330051317	130,097375	1,08121829
65	DES/CBC	1165039741	459,227413	3,81656492	181	SHACAL2/ECB	314054740	123,791954	1,02881495
66	DES/CFB-FIPS	1159563528	457,068837	3,79862534	182	SHACAL2/OFB	312073011	123,010809	1,02232299
67	DES/CFB	1115348380	439,640412	3,65378051	183	SHARK/CBC-CTS	1224942616	482,839519	4,01280132
68	DES/CTR	1144126264	450,983881	3,74805426	184	SHARK/CBC	1171571323	461,801987	3,83796179
69	DES/ECB	1129543584	445,23578	3,70028272	185	SHARK/CFB-FIPS	1186720883	467,773538	3,88759038
70	DES/OFB	1129263312	445,125304	3,69936457	186	SHARK/CFB	1125085514	443,478529	3,68567848
71	3DES 64/CBC-CTS	1425119373	561,743827	4,66856229	187	SHARK/CTR	1119799603	441,394964	3,66836231
72	3DES 64/CBC	1296674637	511,114358	4,24778894	188	SHARK/ECB	1124731061	443,338813	3,68451733
73	3DES 64/CFB-FIPS	1170250039	461,281173	3,83363338	189	SHARK/OFB	1107759752	436,649178	3,62892085
74	3DES 64/CFB	1121609614	442,108422	3,67429175	190	SKIPJACK/CBC-CTS	1539528919	606,841001	5,04335763
75	3DES 64/CTR	1265923210	498,99297	4,14705005	191	SKIPJACK/CBC	1385995593	546,322283	4,54039632
76	3DES 64/ECB	1259200941	496,343232	4,1250285	192	SKIPJACK/CFB-FIPS	1185700484	467,371324	3,88424765
77	3DES 64/OFB	1260250366	496,756887	4,12846632	193	SKIPJACK/CFB	1125280320	443,555316	3,68631665
78	3DES 124/CBC-CTS	1425264871	561,801179	4,66903893	194	SKIPJACK/CTR	1336933439	526,983298	4,37967314
79	3DES 124/CBC	1297280228	511,353066	4,2497728	195	SKIPJACK/ECB	1339439071	527,970951	4,38788136
80	3DES 124/CFB-FIPS	1170930294	461,549311	3,83586184	196	SKIPJACK/OFB	1324718614	522,168541	4,33965848
81	3DES 124/CFB	1119430969	441,249659	3,6671547	197	ThreeWay/CBC-CTS	808888607	318,84219	2,64984599
82	3DES 124/CTR	1266408190	499,184136	4,1486388	198	ThreeWay/CBC	780493235	307,649496	2,55682531
83	3DES 124/ECB	1258165174	495,934961	4,12163542	199	ThreeWay/CFB-FIPS	817382862	322,190397	2,6776724
84	3DES 124/OFB	1260302720	496,777524	4,12863783	200	ThreeWay/CFB	704064274	277,52325	2,30645094
85	3DES 196/CBC-CTS	1422386359	560,666546	4,65960919	201	ThreeWay/CTR	753304821	296,932553	2,46775852
86	3DES 196/CBC	1293831439	509,993646	4,23847488	202	ThreeWay/ECB	801291337	315,84755	2,62495802
87	3DES 196/CFB-FIPS	1169248259	460,886298	3,83035164	203	ThreeWay/OFB	752097959	296,45684	2,46380495
88	3DES 196/CFB	1118041184	440,701843	3,66260189	204	Twofish 128/CBC-CTS	680273464	268,145551	2,2285144
89	3DES 196/CTR	1264289047	498,348827	4,14169668	205	Twofish 128/CBC	621711167	245,061864	2,03666961
90	3DES 196/ECB	1256835956	495,411019	4,11728102	206	Twofish 128/CFB-FIPS	624108085	246,006665	2,04452169
91	3DES 196/OFB	1259510512	496,465257	4,12604263	207	Twofish 128/CFB	531939798	209,676399	1,74258671
92	GOST/CBC-CTS	1256178794	495,151983	4,11512822	208	Twofish 128/CTR	629301586	248,053804	2,06153513
93	GOST/CBC	1196319626	471,557105	3,919035	209	Twofish 128/ECB	593207212	233,82637	1,94329322
94	GOST/CFB-FIPS	1161225792	457,724057	3,80407077	210	Twofish 128/OFB	588153075	231,834165	1,92673633
95	GOST/CFB	1112689933	438,592523	3,64507168	211	Twofish 192/CBC-CTS	681409482	268,593339	2,23223589
96	GOST/CTR	1157969470	456,440502	3,79340335	212	Twofish 192/CBC	621741911	245,073983	2,03677032
97	GOST/ECB	1151180516	453,764478	3,77116335	213	Twofish 192/CFB-FIPS	623492836	245,76415	2,04250619
98	GOST/OFB	1145484127	451,519115	3,75250249	214	Twofish 192/CFB	532483797	209,890829	1,7443688
99	IDEA/CBC-CTS	1214863664	478,866667	3,97978359	215	Twofish 192/CTR	628895017	247,893545	2,06020325
100	IDEA/CBC	1163246727	458,520655	3,81069117	216	Twofish 192/ECB	593328960	233,87436	1,94369206
101	IDEA/CFB-FIPS	1164521977	459,023324	3,81486877	217	Twofish 192/OFB	588571737	231,99919	1,92810783
102	IDEA/CFB	1115858143	439,841346	3,65545045	218	Twofish 256/CBC-CTS	682089101	268,861226	2,23446226
103	IDEA/CTR	1129994742	445,413615	3,70176067	219	Twofish 256/CBC	622264341	245,270911	2,03848175
104	IDEA/ECB	1122307348	442,38345	3,67657746	220	Twofish 256/CFB-FIPS	623374966	245,717689	2,04212006
105	IDEA/OFB	1120631515	441,722882	3,67108758	221	Twofish 256/CFB	531734450	209,595456	1,74191401
106	MARS/CBC-CTS	664738655	262,022146	2,17762377	222	Twofish 256/CTR	628039800	247,556442	2,05740164
107	MARS/CBC	610477999	240,634051	1,99987076	223	Twofish 256/ECB	593047440	233,763392	1,94276983
108	MARS/CFB-FIPS	620971252	244,77021	2,03424571	224	Twofish 256/OFB	586559019	231,205831	1,92151434
109	MARS/CFB	528175419	208,192582	1,73025495	225	XTEA/CBC-CTS	1233570977	486,240588	4,04106706
110	MARS/CTR	622740372	245,467549	2,04004119	226	XTEA/CBC	1176315221	463,671905	3,85350237
111	MARS/ECB	584574786	230,423699	1,91501418	227	XTEA/CFB-FIPS	1166269202	459,712034	3,82059252
112	MARS/OFB	582482490	229,598972	1,90816	228	XTEA/CFB	1116502634	440,095388	3,65756174
113	RC2/CBC-CTS	1309363893	516,116122	4,2893578	229	XTEA/CTR	1141324117	449,87935	3,73887468
114	RC2/CBC	1234525254	486,616738	4,04419318	230	XTEA/ECB	1139072728	448,991913	3,73149333
115	RC2/CFB-FIPS	1166127511	459,656183	3,82012835	231	XTEA/OFB	1137706995	448,453578	3,72702531
116	RC2/CFB	1118773470	440,99049	3,6650008	-	-	-	-	-

Table 7: The unity and signification for each parameters

Parameter	Unity	Signification
Break Probability ρ	-	
Resistance Factor S	Bit	
Resistance Factor Efficacy ES	Bit	
Key-Block Factor KB	-	
Total Resistance Factor for Cipher Structure TS	RB	Round bit
Index of Security IS	SC	Security per cipher
W	-	
Best Success Rate Factor BA	-	
Index of Risk IR	$RC = SC^{-1}$	Risk per cipher
Complexity	Cycle	CPU cycle
Normalize Complexity Γ	Xcycle	CPU cycle per Xor
$\underline{\Gamma}$	CP	Complexity per Processor
Cipher Cost CC	PR	Performance cost for risk and complexity

Blowfish and Twofish less costly in use than AES with 256/192/128 bits in key. Furthermore, this absence of risk make Twofish with 128 bits in key more optimal since it requests less time for encryption. After the AES, we notice that SHACAL2 and Serpent with 192 bits in key come next, followed by XTEA, SHARK, IDEA, Camellia with 128 bit in key, Camellia with 256 bits in key, Serpent with 256 bits in key, CAST with 256 bits in key, MARS, RC6 and CAST with 128 bits in key. Finally, in the sorted CC list, we note that the greatest cost for using a cipher was taken by DES, followed by 3DES, RC5, RC2 and SKIPJACK.

This result has the advantage of combining theoretical (cryptanalysis attack) and experimental (complexity) results. This combination makes the result valuable and very interesting because a lot of cryptographic studies separate the theoretical background from the experimental results. This separation may cause a loss of information, which makes any comparison between ciphers in their mode of operations less fair and less equitable.

5 Conclusions and Future Work

This article contains new formulas and a definition of risk analysis factors for ciphers. These formulas take into account security factors, risk factors and the ciphers using-cost, while respecting in each cipher its own structure and properties. These parameters include structure, key space, round number, encryption mechanism, complexity and successful cryptanalysis attacks, *etc.*.

These formulas provide a lot of information to allow future comparison in a fair platform, which will help a decision maker to select the most appropriate cipher for its own system with its QOS recommendation. In addition, the ciphers designer can also benefit from these formulas constructed on MLO because it offers the theoretical quantifiable value to test the encryption process before applying any cryptanalysis attack.

Table 8: CC measurement for several cipher with different mode of operation

ID	Cipher/mode	CC	ID	Cipher/mode	CC	ID	Cipher/mode	CC
1	AES128/CBC-CTS	0,913598	83	3DES64/ECB	46,311394	165	RC6256/CFB	49,326927
2	AES128/CBC	0,866341	84	3DES64/OFB	46,349990	166	RC6256/CTR	42,271908
3	AES128/CFB-FIPS	0,931299	85	3DES124/CBC-CTS	52,418960	167	RC6256/ECB	47,942699
4	AES128/CFB	0,797812	86	3DES124/CBC	47,711889	168	RC6256/OFB	45,151879
5	AES128/CTR	0,880670	87	3DES124/CFB-FIPS	43,064941	169	Serpent128/CBC-CTS	44,914185
6	AES128/ECB	0,819800	88	3DES124/CFB	41,170878	170	Serpent128/CBC	18,401617
7	AES128/OFB	0,827240	89	3DES124/CTR	46,576465	171	Serpent128/CFB-FIPS	16,850546
8	AES192/CBC-CTS	0,684291	90	3DES124/ECB	46,273300	172	Serpent128/CFB	16,547380
9	AES192/CBC	0,648608	91	3DES124/OFB	46,351916	173	Serpent128/CTR	14,188476
10	AES192/CFB-FIPS	0,698142	92	3DES196/CBC-CTS	52,313093	174	Serpent128/ECB	17,126104
11	AES192/CFB	0,596957	93	3DES196/CBC	47,585048	175	Serpent128/OFB	16,215067
12	AES192/CTR	0,659049	94	3DES196/CFB-FIPS	43,003079	176	Serpent192/CBC-CTS	16,114726
13	AES192/ECB	0,614101	95	3DES196/CFB	41,119764	177	Serpent192/CBC	1,502350
14	AES192/OFB	0,619291	96	3DES196/CTR	46,498527	178	Serpent192/CFB-FIPS	1,375221
15	AES256/CBC-CTS	0,338464	97	3DES196/ECB	46,224414	179	Serpent192/CFB	1,352549
16	AES256/CBC	0,320875	98	3DES196/OFB	46,322780	180	Serpent192/CTR	1,160307
17	AES256/CFB-FIPS	0,345115	99	GOST/CBC-CTS	68,973789	181	Serpent192/ECB	1,400095
18	AES256/CFB	0,295108	100	GOST/CBC	65,687065	182	Serpent192/OFB	1,323949
19	AES256/CTR	0,326471	101	GOST/CFB-FIPS	63,760146	183	Serpent256/CBC-CTS	1,316338
20	AES256/ECB	0,303837	102	GOST/CFB	61,095157	184	Serpent256/CBC	15,093283
21	AES256/OFB	0,306319	103	GOST/CTR	63,581349	185	Serpent256/CFB-FIPS	13,808402
22	Blowfish/CBC-CTS	0,000000	104	GOST/ECB	63,208584	186	Serpent256/CFB	13,566295
23	Blowfish/CBC	0,000000	105	GOST/OFB	62,895808	187	Serpent256/CTR	11,644703
24	Blowfish/CFB-FIPS	0,000000	106	IDEA/CBC-CTS	1,945411	188	Serpent256/ECB	14,043636
25	Blowfish/CFB	0,000000	107	IDEA/CBC	1,862755	189	Serpent256/OFB	13,300835
26	Blowfish/CTR	0,000000	108	IDEA/CFB-FIPS	1,864797	190	SHACAL2/CBC-CTS	13,216129
27	Blowfish/ECB	0,000000	109	IDEA/CFB	1,786869	191	SHACAL2/CBC	1,502598
28	Blowfish/OFB	0,000000	110	IDEA/CTR	1,809507	192	SHACAL2/CFB-FIPS	1,359075
29	Camellia128/CBC-CTS	2,439061	111	IDEA/ECB	1,797197	193	SHACAL2/CFB	1,297250
30	Camellia128/CBC	2,223550	112	IDEA/OFB	1,794513	194	SHACAL2/CTR	1,120957
31	Camellia128/CFB-FIPS	2,161228	113	MARS/CBC-CTS	35,413582	195	SHACAL2/ECB	1,363086
32	Camellia128/CFB	1,851554	114	MARS/CBC	32,522876	196	SHACAL2/OFB	1,297022
33	Camellia128/CTR	2,268503	115	MARS/CFB-FIPS	33,081898	197	SHARK/CBC-CTS	1,288837
34	Camellia128/ECB	2,124675	116	MARS/CFB	28,138252	198	SHARK/CBC	45,730335
35	Camellia128/OFB	2,128995	117	MARS/CTR	33,176147	199	SHARK/CFB-FIPS	43,737844
36	Camellia192/CBC-CTS	6,993433	118	MARS/ECB	31,142896	200	SHARK/CFB	44,303417
37	Camellia192/CBC	6,379568	119	MARS/OFB	31,031431	201	SHARK/CTR	42,002407
38	Camellia192/CFB-FIPS	6,183089	120	RC2/CBC-CTS	73,846996	202	SHARK/ECB	41,805070
39	Camellia192/CFB	5,296762	121	RC2/CBC	69,626161	203	SHARK/OFB	41,989174
40	Camellia192/CTR	6,496791	122	RC2/CFB-FIPS	65,768588	204	SKIPJACK/CBC-CTS	41,355590
41	Camellia192/ECB	6,080542	123	RC2/CFB	63,097860	205	SKIPJACK/CBC	86,800675
42	Camellia192/OFB	6,106682	124	RC2/CTR	67,867250	206	SKIPJACK/CFB-FIPS	78,144263
43	Camellia256/CBC-CTS	4,368407	125	RC2/ECB	66,895077	207	SKIPJACK/CFB	66,851360
44	Camellia256/CBC	3,989085	126	RC2/OFB	67,335425	208	SKIPJACK/CTR	63,444791
45	Camellia256/CFB-FIPS	3,870495	127	RC5*/CBC-CTS	59,065524	209	SKIPJACK/ECB	75,378074
46	Camellia256/CFB	3,310211	128	RC5*/CBC	56,995883	210	SKIPJACK/OFB	75,519345
47	Camellia256/CTR	4,062092	129	RC5*/CFB-FIPS	59,004923	211	ThreeWay/CBC-CTS	74,689386
48	Camellia256/ECB	3,799487	130	RC5*/CFB	56,631110	212	ThreeWay/CBC	64,465559
49	Camellia256/OFB	3,813245	131	RC5*/CTR	55,327618	213	ThreeWay/CFB-FIPS	62,202548
50	CAST128*/CBC-CTS	58,993581	132	RC5*/ECB	55,093507	214	ThreeWay/CFB	65,142521
51	CAST128*/CBC	56,342755	133	RC5*/OFB	55,032004	215	ThreeWay/CTR	56,111431
52	CAST128*/CFB-FIPS	57,251350	134	RC5**/CBC-CTS	59,065524	216	ThreeWay/ECB	60,035728
53	CAST128*/CFB	54,767648	135	RC5**/CBC	56,995883	217	ThreeWay/OFB	63,860083
54	CAST128*/CTR	54,994768	136	RC5**/CFB-FIPS	59,004923	218	Twofish128/CBC-CTS	59,939545
55	CAST128*/ECB	54,410741	137	RC5**/CFB	56,631110	219	Twofish128/CBC	0,000000
56	CAST128*/OFB	54,418252	138	RC5**/CTR	55,327618	220	Twofish128/CFB-FIPS	0,000000
57	CAST128**/CBC-CTS	40,527335	139	RC5**/ECB	55,093507	221	Twofish128/CFB	0,000000
58	CAST128**/CBC	38,706274	140	RC5**/OFB	55,032004	222	Twofish128/CTR	0,000000
59	CAST128**/CFB-FIPS	39,330460	141	RC5***/CBC-CTS	59,065524	223	Twofish128/ECB	0,000000
60	CAST128**/CFB	37,624209	142	RC5***/CBC	56,995883	224	Twofish128/OFB	0,000000
61	CAST128**/CTR	37,780236	143	RC5***/CFB-FIPS	59,004923	225	Twofish192/CBC-CTS	0,000000
62	CAST128**/ECB	37,379022	144	RC5***/CFB	56,631110	226	Twofish192/CBC	0,000000
63	CAST128**/OFB	37,384182	145	RC5***/CTR	55,327618	227	Twofish192/CFB-FIPS	0,000000
64	CAST256/CBC-CTS	23,094559	146	RC5***/ECB	55,093507	228	Twofish192/CFB	0,000000
65	CAST256/CBC	21,170467	147	RC5***/OFB	55,032004	229	Twofish192/CTR	0,000000
66	CAST256/CFB-FIPS	20,882287	148	RC6128/CBC-CTS	41,101655	230	Twofish192/ECB	0,000000
67	CAST256/CFB	17,799309	149	RC6128/CBC	38,453726	231	Twofish192/OFB	0,000000
68	CAST256/CTR	21,689633	150	RC6128/CFB-FIPS	40,461357	232	Twofish256/CBC-CTS	0,000000
69	CAST256/ECB	20,327965	151	RC6128/CFB	34,707866	233	Twofish256/CBC	0,000000
70	CAST256/OFB	20,352091	152	RC6128/CTR	39,363245	234	Twofish256/CFB-FIPS	0,000000
71	DES/CBC-CTS	57,862345	153	RC6128/ECB	37,075234	235	Twofish256/CFB	0,000000
72	DES/CBC	54,941989	154	RC6128/OFB	36,898767	236	Twofish256/CTR	0,000000
73	DES/CFB-FIPS	54,683737	155	RC6192/CBC-CTS	47,569456	237	Twofish256/ECB	0,000000
74	DES/CFB	52,598599	156	RC6192/CBC	44,492275	238	Twofish256/OFB	0,000000
75	DES/CTR	53,955732	157	RC6192/CFB-FIPS	46,860443	239	XTEA/CBC-CTS	0,000000
76	DES/ECB	53,268029	158	RC6192/CFB	40,179859	240	XTEA/CBC	1,363389
77	DES/OFB	53,254812	159	RC6192/CTR	45,589593	241	XTEA/CFB-FIPS	1,300107
78	3DES64/CBC-CTS	52,413608	160	RC6192/ECB	42,889572	242	XTEA/CFB	1,289004
79	3DES64/CBC	47,689617	161	RC6192/OFB	42,680038	243	XTEA/CTR	1,234000
80	3DES64/CFB-FIPS	43,039922	162	RC6256/CBC-CTS	50,047571	244	XTEA/ECB	1,261434
81	3DES64/CFB	41,251006	163	RC6256/CBC	46,789653	245	XTEA/OFB	1,258946
82	3DES64/CTR	46,558629	164	RC6256/CFB-FIPS	46,311394	-	-	-

Cipher specification: 1) CAST-128*/CAST-128** is for key length from 40 to 80/80 to 128 bits; 2) RC5*/RC5**/RC5*** is for plaintext length 32/64/128 bits.

Moreover, these formulas are developed so that their value can be taken as a standard, since even when the system or machine, OS, CPU, *etc.* changes, the result is not very much affected (change that does not exceed $0,3\%$). Our future work will concern two paths:

- The first will focus on obtaining more ciphers or algorithms using-cost measurement.
- The second will concentrate on getting deeper in risk analysis study over cipher.

Acknowledgments

The authors are grateful to the anonymous reviewers for valuable comments.

References

[1] E. Biham and A. Shamir, "Differential cryptanalysis of DES-like cryptosystems," *Journal of Cryptography*, vol. 4, no. 1, pp. 3–72, 1991.

[2] A. Biryukov, "Meet-in-the-middle attack," in *Encyclopedia of Cryptography and Security*, pp. 772–773, 2011.

[3] A. Biryukov, D. Khovratovich, and I. Nikolić, "Distinguisher and related-key attack on the full AES-256," in *Advances in Cryptology (CRYPTO'09)*, pp. 231–249, 2009.

[4] A. Biryukov and D. Wagner, "Slide attacks," in *International Workshop on Fast Software Encryption*, pp. 245–259, 1999.

[5] C. Blondeau, *La Cryptanalyse Différentielle Et Ses Généralisations*, Thesis, Université Pierre et Marie Curie-Paris VI, Nov. 2011.

[6] C. Cid and G. Leurent, "An analysis of the XSL algorithm," in *International Conference on the Theory and Application of Cryptology and Information Security*, pp. 333–352, 2005.

[7] S. Dey and R. Ghosh, "A review of cryptographic properties of S-Boxes with generation and analysis of Crypto secure S-Boxes," *International Journal of Electronics and Information Engineering*, vol. 8, no. 1, pp. 49-73, 2018.

[8] I. R. Dragomir and M. Lazăr, "Generating and testing the components of a block cipher," in *8th International Conference on Electronics, Computers and Artificial Intelligence*, pp. 1–4, June 2016.

[9] A. B. Forouzan, *Data Communications and Networking*, McGraw-Hill Forouzan networking series, McGraw-Hill Higher Education, 2007.

[10] W. F. Friedman, *The Index of Coincidence and Its Applications in Cryptanalysis*, Aegean Park Press California, 1987.

[11] B. Gérard, *Statistical Cryptanalyses of Symmetric-Key Algorithms*, Thesis, Université Pierre et Marie Curie-Paris VI, Dec. 2010.

[12] Y. Harmouch and R. E. Kouch, "A fair comparison between several ciphers in characteristics, safety and speed test,". in *Europe and MENA Cooperation Advances in Information and Communication Technologies*, pp. 535–547, 2017.

[13] V. T. Hoang and P. Rogaway, "On generalized feistel networks," in *Annual Cryptology Conference*, pp. 613–630, 2010.

[14] M. S. Hwang, C. T. Li, J. J. Shen, Y. P. Chu, "Challenges in e-government and security of information", *Information & Security: An International Journal*, vol. 15, no. 1, pp. 9–20, 2004.

[15] J. Kelsey, B. Schneier, and D. Wagner, "Mod n cryptanalysis, with applications against RC5P and M6," in *International Workshop on Fast Software Encryption*, pp. 139–155, 1999.

[16] M. C. Lee, "Information security risk analysis methods and research trends: Ahp and fuzzy comprehensive method," *International Journal of Computer Science and Information Technology*, vol. 6, pp. 29–45, Mar. 2014.

[17] M. Matsui, "Linear cryptanalysis method for des cipher," in *Workshop on the Theory and Application of of Cryptographic Techniques*, pp. 386–397, 1993.

[18] H. H. Ngo, X. Wu, P. D. Le, C. Wilson, and B. Srinivasan, "Dynamic key cryptography and applications," *International Journal of Network Security*, vol. 10, pp. 161–174, May 2010.

[19] H. Sato, "A new formula of security risk analysis that takes risk improvement factor into account," in *IEEE Third International Conference on Privacy, Security, Risk and Trust and 2011 IEEE Third International Conference on Social Computing*, pp. 1243–1248, Oct. 2011.

[20] N. Shukla and S. Kumar, "A comparative study on information security risk analysis practices," in *Issues and Challenges in Networking, Intelligence and Computing Technologies (ICNICT'12)*, pp. 28–33, Nov. 2012.

[21] O. Tornea, *Contributions to DNA Cryptography: Applications to Text and Image Secure Transmission*, PhD Thesis, Université Nice Sophia Antipolis, 2013.

[22] D. Wagner, "The boomerang attack," in *International Workshop on Fast Software Encryption*, pp. 156–170, 1999.

[23] C. Wang and W. A. Wulf, "Towards a framework for security measurement," in *20th National Information Systems Security Conference, Baltimore*, pp. 522–533, 1997.

[24] J. Wang, K. Fan, W. Mo, and D. Xu, "A method for information security risk assessment based on the dynamic bayesian network," in *International Conference on Networking and Network Applications*, pp. 279–283, July 2016.

[25] Y. Yeom, "Integral cryptanalysis and higher order differential attack," *Trends in Mathematics*, vol. 8, no. 1, pp. 101–118, 2005.

[26] A. Yun, J. H. Park, and J. Lee, "On lai–massey and quasi-feistel ciphers," *Designs, Codes and Cryptography*, vol. 58, no. 1, pp. 45–72, 2011.

Appendix-A: The Computation of Break Probability for Each Structure and Operation

Let us consider two random variables X and K with i resp. j is a number while p and q are probabilities. Now, assume that the distribution probability is given by:

$$Pr(X) = \begin{cases} p, & X = i \\ 1-p, & X \neq i \end{cases} and \quad Pr(K) = \begin{cases} q, & K = j \\ 1-q, & K \neq j \end{cases}$$

X and K are two independent variables, Thus:

$$Pr(X,K) = \begin{cases} pq, & X = i, K = j \\ p(1-q), & X = i, K \neq j \\ q(1-p), & X \neq i, K = j \\ (1-p)(1-q), & X \neq i, K \neq j \end{cases}$$

- **Left/Right rotation**:

  Let us consider f:$\{0,1\}^n \times GF(2^8) \rightarrow \{0,1\}^n$ f can be described as $f(\xi,\phi) \rightarrow \xi'$ where f is a function, ξ is a binary-vector, ϕ is a number with $dim\xi > \phi$ and ξ' is the binary-vector results. We denote by $dim\xi$ the size of the vector ξ.

  Given f, we have $f\ (\xi,\phi) = \xi \begin{pmatrix} \ll \\ or \\ \gg \end{pmatrix} \phi$ where "$\ll$" resp. "$\gg$" indicates left resp. right rotation. Of note, f is itself invertible with $Pr(f=1) = 1/\phi$ because ξ' has ϕ possibilities. ϕ is unknown, hence $Pr(f=1) = \frac{1}{dim\xi - 1}$

- **NOT**:

  Let us consider f:$\{0,1\} \rightarrow \{0,1\}$ where f is the bitwise NOT function. For such function, we have $f(x) = \overline{x}$ with x is a binary variable. The probability of guessing the result is equal to $Pr(f=1) = p + (1-p) = 1$, thus, for the general case (binary vector) we have $Pr(f^n) = \prod_{i=1}^{n}[p + (1-p)] = 1$.

- **Conditional NOT**:

  Let us consider f:$\{0,1\}^2 \rightarrow \{0,1\}$ where f is the bitwise conditional-not-function. Given f, we write f(x,k)=y with x, k and y $\in \{0,1\}$. To show y, let us consider that f applies "not" to x if k is true, so we have $f(x,k) = \overline{x}k + x\overline{k}$ where "+" indicates logical addition. As observed, f is equivalent to XOR function, so Conditional Not and XOR have the same probabilities (see below for more details).

- **AND**:

  Let us consider f:$\{0,1\}^2 \rightarrow \{0,1\}$ where f is the bitwise AND function. For such function, we have $f(X,K) = X \times K$ where "$\times$" indicates "AND" and X, K $\in \{0,1\}$. Note that f is not invertible, this implies that even if X is found, K cannot be known (vice versa). Consequently, the only possible case of breaking f is to know both X and K. Therefore $Pr(f=1) = Pr(X,K) = pq = 1/2^2$. As for the general case (X and K are binary vector) we have $Pr(f^n) = Pr(X,K) = (\prod_{i=1}^{n} \frac{1}{\#f})^2 = 1/2^{2n}$ with # denotes the set cardinal.

- **OR**:

  Let us consider f:$\{0,1\}^2 \rightarrow \{0,1\}$ where f is the bitwise OR function. For such function, we have $f(X,K) = X + K$ where "+" indicates "OR" and X, K $\in \{0,1\}$. Remark that f is not invertible, this implies that even if X is found, K cannot be known (vice versa). Consequently, the only possible case of breaking f is to know both X and K. thus $Pr(f=1) = Pr(X,K) = pq = 1/2^2$. As for the general case (X and K are binary vector) we have $Pr(f^n) = Pr(X,K) = (\prod_{i=1}^{n} \frac{1}{\#f})^2 = 1/2^{2n}$.

- **XOR**:

  Let us consider f:$\{0,1\}^2 \rightarrow \{0,1\}$ where f is the bitwise XOR function. For such function, we have $f(X,K) = X + K$ where "+" indicates mod2 addition and X, K $\in \{0,1\}$. Given these two variables, f can only present one of the following two scenarios: f:$\{0,1\}^2 \rightarrow \{0\}$ is a linear expression and is equivalent to X = K and f:$\{0,1\}^2 \rightarrow \{1\}$ is an affine expression and is equivalent to X≠K. Since $Pr(X=K) = Pr(X=0,K=0) + Pr(X=1,K=1)$ and $Pr(X \neq K) = Pr(X=0,K=1) + Pr(X=1,K=0)$, we have $Pr(X=K) = pq + (1-p)(1-q)$ and $Pr(X=K) = p(1-q) + q(1-p)$. Moreover, f is invertible. This implies that knowing one variable from those defined above led to know the second. Therefore the probability of breaking f is Pr(X|K). This can be solved by: Pr(X|K)=$\frac{Pr(X \bigcap K)}{Pr(K)} = \frac{Pr(X) \times Pr(K)}{Pr(K)} = Pr(X) = p = 1/2$. for the general case (X and K are binary vector) we have $Pr(f^n) = Pr(X,K) = \prod_{i=1}^{n} 1/2 = 1/2^n$.

- **Concatenation**:

  Let us consider f:$\{0,1\}^2 \rightarrow \{0,1\}$ where f is the concatenation function. For such function, we have $f(X,K) = X \| K$ where "$\|$" indicates the concatenation-operation and X, K $\in \{0,1\}$. The probability of guessing X and K from f is equal to Pr(f)=p+q=1, hence for the general case is equal to $Pr(f^n) = Pr(f) = 1$. If X and K were a binary vectors with unequal or unknown size then we will have $Pr(f) = \frac{1}{\#f\text{-}1}$

- **Modular addition**:

  The result proved in XOR can be generalized to modular addition since XOR is mod 2 addition case. Thus, for f defined as f:$\{0,1,\cdots,\xi-1\}^2 \rightarrow$

$\{0,1,\cdots,\xi-1\}$ the break probability is equal to $Pr(f^n) = Pr(X,K) = \prod_{i=1}^{n} \frac{1}{\#f} = 1/\xi^n$.

- **Modular subtraction**:

The result proved in modular addition is the same as modular subtraction since "+" and "?" has the same break probabilities, thus for f defined as f:$\{0,1,\cdots,\xi-1\}^2 \rightarrow \{0,1,\cdots,\xi-1\}$ the break probability is equal to $Pr(f^n) = Pr(X,K) = 1/\xi^n$.

- **Modular multiplication**:

Let us consider f:$GF(\xi)^2 \rightarrow GF(\xi)$ where f is the modular multiplication function and ξ is the modulus. For such function, we have $f(X,K) = X \times K$ where "$\times$" indicates multiplication modξ and X, K $\in GF(\xi)$. Notice that f is not itself invertible, it implies that to find X, K should have a modular multiplicative inverse K'. i.e. $K \times K' \equiv 1(mod\xi)$. Consequently, two scenarios are possible: either K admits a modular multiplicative inverse thus K and ξ are coprime or K do not admits a modular multiplicative inverse. These scenarios shows that the found of one variable X or K do not help of guessing the other one. Thus, the only possible case to break f is Pr(X,K)=pq. As so, for the general case, we have $Pr(f^n) = (\prod_{i=1}^{n} \frac{1}{\#f})^2 = 1/\xi^{2n}$.

- **Modular exponentiation**:

The modular exponentiation is a special case of the modular multiplication where knowing both X and K is the only way to break the operation, thus $Pr(f^n) = 1/\xi^{2n}$.

- **P-box**:

Let us consider f:$\{0,1\}^n \rightarrow \{0,1\}^n$ where f is a permutation function (P-box). Given f, we write $f(x_0,x_1,\cdots,x_{n-1})$=$(x_i,\cdots,x_j,\cdots,x_k)$ where i, j, k $\in [0,n-1]$. If we consider f as a black box (dynamic P-box) where the linear link between input and output is not known, the breaking probability for f for a binary vector X is $Pr(f) = Pr(X) = \prod_{i=1}^{n-1} \frac{1}{p} = \prod_{i=1}^{n-1} \frac{1}{\#f} = 1/2^{n-1}$. As for the static P-box where the linear link between input and output is exactly known, we have $Pr(f) = Pr(X) = \prod_{i=1}^{n-1} \frac{1}{p(1-p)} = 1$.

- **S-box**:

Let us consider f:$GF(\xi)^n \rightarrow GF(\xi)^m$ where f is a substitution function and 2 is the modulus. Given f, we write $f(x_0,x_1,\cdots,x_{n-1})$=$(y_0,y_1,\cdots,y_{n-1})$. For instance, the AES S-box is written as $f(x_i)$=$\sum_{u\in GF(2)^n} a_u \prod_{i=1}^{n} x_i^{u_i}$, $a_u \in GF(2)^n$.
Thereby, this equation can be denoted as $f(x_i)$=$(l \circ h)$, where "l" indicates the n×m binary matrix and "h" is a function. For example, h(x) in AES is equal to h(x)= $\begin{cases} x^{-1}, & X \neq 0 \\ 0 \quad , & X = 0 \end{cases}$
Thus, as shown by Liam Keliher in "Linear Cryptanalysis of Substitution-Permutation Networks" in ch.4, the probability for breaking the S-box is Pr(f)=$\frac{1}{2^n-1}$.

- **Feistel**:

Let us consider f:$\{0,1\}^n \times \{0,1\}^m \rightarrow \{0,1\}^n$ where f is a Feistel function and m<n. Given f, we write f(X,K) with X, K are two binary vector and f(X,K)=$\begin{cases} x_{m+i}, & 1 \leqslant i \leqslant m \\ x_{i-m} \oplus G(x_i, k_{i-m}), & m < i \leqslant n \end{cases}$ with G is a round function.
Since f admits a linear liaison for n-m random binary variables, the security for this structure is built over K, and the only possible case to break f is by guessing K, as so Pr(f)=pq+q(1-p)=q=$\prod_{i=1}^{m} \frac{1}{2} = 1/2^m$. It must be mention that in the case of m=n/2 the Feistel structure is called balanced Feistel function, otherwise, it is called unbalanced Feistel function and the probability turn to be equal to Pr(f)=$\frac{1}{2^{n-m}}$.

- **Lai-Massey**:

Let us consider f:$\{0,1\}^n \times \{0,1\}^m \rightarrow \{0,1\}^n$ where f is a Lai-Massey function and m<n. Given f, we write f(X,K) with X, K are two binary vector and f(X,K)=$\begin{cases} \sigma(x_i + G(x_i - x_{\frac{n}{2}+i}, k_j)) \\ x_{\frac{n}{2}+i} + G(x_i - x_{\frac{n}{2}+i}, k_j) \end{cases}$ $1 \leqslant i \leqslant \frac{n}{2}$ and $1 \leqslant j \leqslant m$.
G is a round function and σ is an orthomorphism permutation (in mathematical sense, that is, a bijection not a P-box). The Lai-Massey schema differs from Feistel schema, because it modifies both the left half and the right half of the plaintext block. Thus the security for this structure is built over K and P. Therefore the only possible case to break f is by guessing either X or K, as so Pr(f)=$\prod_{i=1}^{n} \frac{1}{2}[q(1-p)+p(1-q)] = \prod_{i=1}^{n} \frac{1}{2}[p+q-2pq]$ and since p=q=1/2 $\Rightarrow$ Pr(f)=$\prod_{i=1}^{n} \frac{1}{2} = 1/2^n$.

Biography

Youssef Harmouch is a Ph.D. student at National Institute of Post & Telecommunication INPT-Rabat Morocco. He started his career in 2012 as a network and telecommunications engineer Specialized in VOIP and information security. Since 2014, he returned to INPT as a PhD candidate working in fields of cryptography within "Multimedia, Signal And Communication Systems" Laboratory, where he focus on cryptographic schema, cipher design, cryptanalysis Study, risk analysis, advanced mathematical theory precisely in algebra, chaos and coding theory.

Rachid Elkouch is a Professor and the Manager of PABX Laboratory, attached to Systems and Communications Department at INPT Since 1981, he started his carrier as a telecommunications engineer, and then in 1989 he got his Master's degree of Science in Telecommunications from the University of Colorado-Boulder in the USA. In 2005, he obtained his doctorate degree

in applied mathematical from Mohammed Ben Abdellah University-Faculty of Sciences and Techniques of Fez-USMBA. Meanwhile, he continued working as an assistant for the INPT Electricity and electronics laboratory for five years since 1981. Between 1990 and 1992, he became a specialized tutor for a number of technical inspectors. More specifically, it was a technological transfer program within the framework of a convention between France & Morocco. Since 1992, he was in charge of the Engineering Cycle Internship and then officially the Manager of the Internships Service since 1998. He was then promoted to the position of Deputy Director of Internships and International Relations between the INPT and outside companies since 2008. In 2010, he was assigned to the position of Deputy Director of Continuing Education. He is a member of the research team Phare (UFR MDA) from the Laboratory of Computer Science and Mathematics from USMBA, a tutor of 2 modules on the E-platform Miage (Lyon 1): B202-Basics of Telecommunications and B214-Networks Protocols.

Hussain Ben-azza is a Professor at ENSAM-National High School of Arts and Trades, Meknes, Morocco, attached to Department of Industrial and Production Engineering, Moulay Ismail University. He obtained his Ph.D. degree in mathematics and computer science in 1995 from Claude Bernard University Lyon 1, France. His research interests include coding theory, cryptography, wireless communications, but also applications of optimization techniques to industrial engineering.

A Novel Dual Image-based High Payload Reversible Hiding Technique Using LSB Matching

Yu-Lun Wang[1], Jau-Ji Shen[1] and Min-Shiang Hwang[2,3]
(Corresponding author: Min-Shiang Hwang)

Department of Management Information Systems, National Chung Hsing University[1]
145 Xingda Rd., South Dist., Taichung City 402, Taiwan
Department of Computer Science and Information Engineering, Asia University[2]
No. 500, Lioufeng Raod, Wufeng Shiang, Taichung 41354, Taiwan, R.O.C.
Department of Medical Research, China Medical University Hospital, China Medical University[3]
No.91, Hsueh-Shih Road, Taichung 40402, Taiwan
(Email: mshwang@asia.edu.tw)
(Received Dec. 12, 2017; revised and accepted Apr. 6, 2018)

Abstract

A dual image technique has already become more and more popular because of easily achieving higher capacity and lower distortion. Dual image copy into two images depends on both images to correctly recover the original images. This paper proposed a novel reversible data hiding to hide the secret information into the least significant bits. The experiment result shows that our proposed method is effective and with high payload.

Keywords: Data Hiding; LSB Matching; High Payload

1 Introduction

With the advance of the technology, the speed of the internet becomes faster and faster, and multimedia spreads more easily [2,13]. Traditional, if we want to send an important message to the receiver, we can encrypt the message into ciphertext; and then the receiver can decrypt the ciphertext by using the key [11,12,15,21]. Nowadays, the copyright of the image become more and more important, and through hiding data in the image we can prove the ownership of the image [22].

Data hiding has two categories, reversible data hiding [17, 18, 20] and non-reversible data hiding [1]. Non-reversible data hiding usually has higher capacity and less distortion because it don't need the recover mechanism [25]. In some case we need reversible data hiding mechanism such as binary images [26]. In some case we need reversible data hiding mechanism such as medical images [5–8, 10, 14]. We can hide the diagnosis of the patient into the X-ray image. However, a little bit of the distortion of the image may cause a diagnostic error, so correctly recovering the original image is necessary. Therefore, hiding information in the image is a popular field of data security. In this field, we pursue higher capacity and less distortion after embedding the secret information.

There are many data hiding schemes which are based on the pixel value differencing method [16, 19] and based on SMVQ [3, 4]. In 2018, Wang *et al.* introduced a survey of reversible data hiding for VQ-compressed images [24]. In 2016, Jana proposed a dual image based reversible data hiding scheme using weighted matrix [9]. In 2015, Lu *et al.* proposed dual imaging-based reversible hiding technique using LSB matching [17]. They select two pixels as a pair (block) and choose the same images to embed. In 2017, Wang *et al.* proposed an improved scheme of Lu *et al*'s scheme to increase the capacity [23]. In this paper, we will propose an improvement of Wang *et al.*'s scheme to increase the capacity and PSNR.

This paper will be described as follow. Section 2 will introduce the proposed method. Section 3 will show our experiment result and analysis of the proposed scheme. Finally, Section 4 will make a conclusion of this paper.

2 The Proposed Scheme

In our proposed scheme, we first copy the cover image into two copies, and use both copies to embed the secret bits. Then in the first image, we transform the value of the pixel into binary pattern and change its LSB (1) and LSB (2) into the secret bits. In the second image, we do the same work like the first image and change its LSB (3) into secret bits. Figure 1 shows the embedding phase in our proposed scheme.

In the extraction and recover phase, we can easily ex-

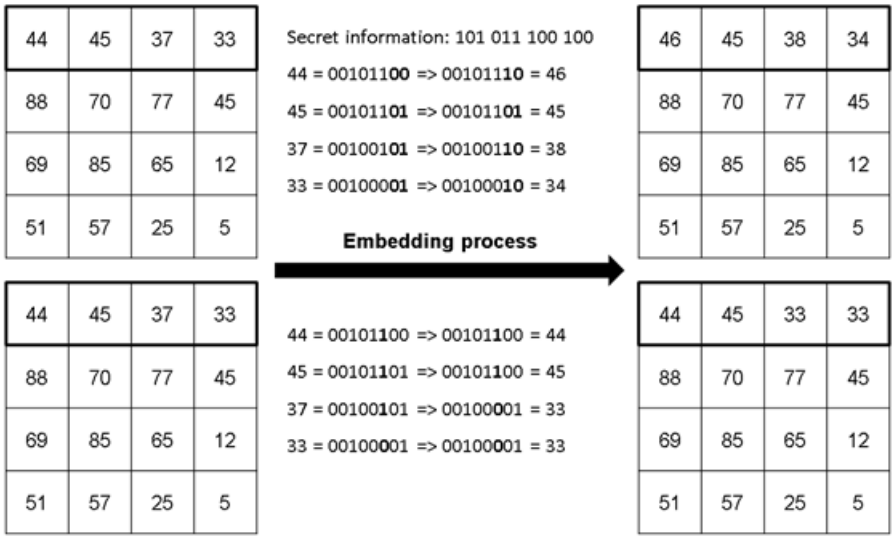


Figure 1: The embedding phase example of our proposed scheme

tract the secret bits by the rule of embedding phase. We can extract the LSB (1) and LSB (2) in the first image and LSB (3) in the second image. Then we can use the LSB (3) of the first image and the LSB (1) and LSB (2) of the second image to correctly recover the original image. Figure 2 shows the extraction and recover phase in our proposed scheme.

3 Analysis and Experiments

There are two criteria in data hiding area, quality and capacity. We use a peak signal-to-noise ratio (PSNR) to quantify image quality as follows:

$$MSE = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{(} j=0)^{(}n-1)[P_{(i,j)}-X_{(i,j)}]^2$$

$$PSNR = 10\times\log_1 0[\frac{255^2}{MSE}].$$

Here, m and n are the images sizes, P is the original image, and X is the image after embedding. The size of the images is 512×512, and the secret information we assume is all of 1. The results in Table 1 show that the capacity of the proposed method is high payload and less distortion.

Through the Table 1 we can make sure that our proposed scheme have very high capacity and acceptable distortion. Chang *et al.*'s scheme [1] is an irreversible scheme and Lu *et al.*'s scheme [17] is a reversible scheme.

4 Conclusion

We propose a high payload data hiding scheme with less distortion. The embedding phase, extraction and recover phase are very easy and effective. Nevertheless, the experiment also shows that our method have an acceptable distortion results and higher capacity. The capacity of this method can be stable at 1.5 bpp.

Acknowledgment

This research was partially supported by the Ministry Of Science and Technology, Taiwan (ROC), under contract no.: MOST 103-2632-E-324-001-MY3.

References

[1] W. J. Chen, C. C. Chang, T. Hoang Ngan Le, 'High payload steganography mechanism using hybrid," *Expert Systems with Applications*, vol. 37, pp. 3292–3301, 2010.

[2] H. W. Chien, S. C. Liao, S. L. Huang, C. M. Chang, H. L. Chen and H. T. Chu, "Selecting internet videos and pictures for personalized reminiscence therapy," *International Journal of Electronics and Information Engineering*, vol. 5, no. 1, pp. 47–55, 2016.

[3] S. F. Chiou, I-En Liao, and M. S. Hwang, "A capacity-enhanced reversible data hiding scheme based on SMVQ", *Imaging Science Journal*, vol. 59, no. 1, pp. 17–24, Feb. 2011.

[4] S. F. Chiou, Y. C. Lu, I-En Liao, and M. S. Hwang, "An efficient reversible data hiding scheme based on SMVQ," *Imaging Science Journal*, vol. 61, no. 6, pp. 467–474, 2013.

[5] L. C. Huang, M. S. Hwang, L. Y. Tseng, "Reversible and high-capacity data hiding in high quality medical images," *KSII Transactions on Internet and Information Systems*, vol. 7, no. 1, pp. 132–148, 2013.

[6] L. C. Huang, M. S. Hwang, and L. Y. Tseng, "Reversible data hiding for medical images in cloud computing environments based on chaotic Henon

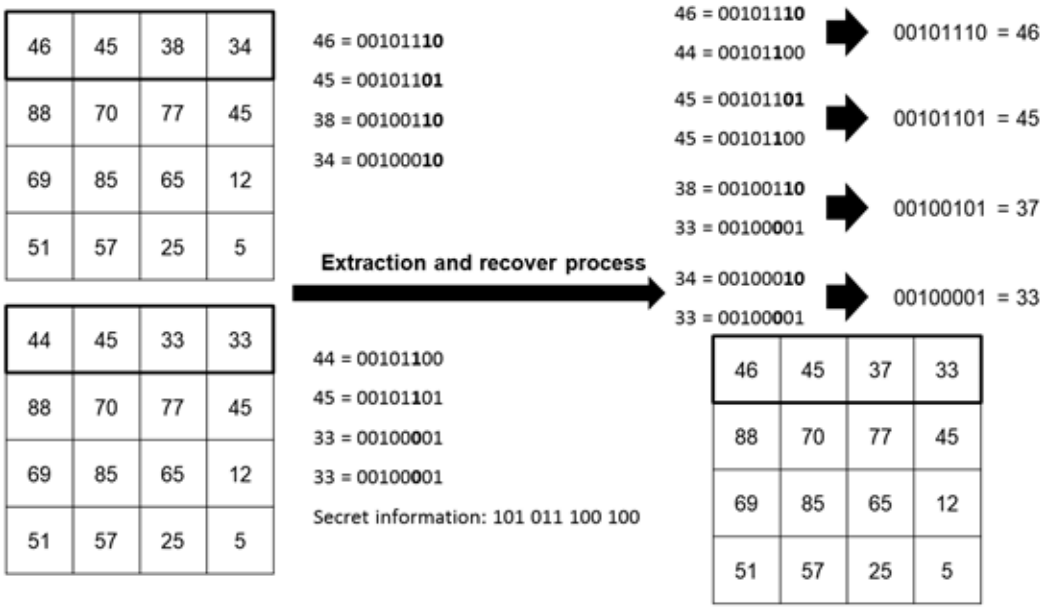


Figure 2: The extraction and recover phase of our proposed scheme

Table 1: The image and total hidden capacity comparison table

Schemes	PSNR/Capacity	Lena	Peppers	Mandrill	Boat	Goldhill
The Proposed Scheme	PSNR (1)	42.70	42.70	42.70	42.77	42.71
	PSNR (2)	38.09	39.15	39.11	39.04	39.01
	Capacity	786432	786432	786432	786432	786432
Chang *et al.*'s Scheme [1]	PSNR (1)	39.89	39.94	39.91	39.89	39.90
	PSNR (2)	39.89	39.94	39.91	39.89	39.90
	Capacity	802895	799684	802524	802716	802698
Lu *et al.*'s Scheme [17]	PSNR (1)	49.13	49.11	47.95	49.00	49.17
	PSNR (2)	49.12	49.08	49.15	49.07	49.09
	Capacity	524288	524192	522996	524208	524288
Wang *et al.*'s Scheme [23]	PSNR (1)	40.97	40.99	40.94	40.96	40.98
	PSNR (2)	41.30	41.23	41.34	41.56	41.24
	Capacity	617088	608877	618977	632797	613288

map," *Journal of Electronic Science and Technology*, vol. 11, no. 2, pp. 230–236, 2013.
[7] L. C. Huang, L. Y. Tseng, M. S. Hwang, "The study on data hiding in medical images", *International Journal of Network Security*, vol. 14, no. 6, pp. 301–309, 2012.
[8] L. C. Huang, L. Y. Tseng, M. S. Hwang, "A reversible data hiding method by histogram shifting in high quality medical images", *Journal of Systems and Software*, vol. 86, no. 3, pp. 716–727, Mar. 2013.
[9] B. Jana, "Dual image based reversible data hiding scheme using weighted matrix," *International Journal of Electronics and Information Engineering*, vol. 5, no. 1, pp. 6–19, 2016.
[10] B. Jana, D. Giri and S. K. Mondal, "Dual-image based reversible data hiding scheme using pixel value difference expansion," *International Journal of Network Security*, vol. 18, no. 4, pp. 633–643, 2016.
[11] F. Jiang, P. Salama, B. King, "A public-key approach of selective encryption for images," *International Journal of Network Security*, vol. 19, no. 1, pp. 118–126, 2017.
[12] C. Jin, H. Liu, "A color image encryption scheme based on Arnold scrambling and quantum chaotic," *International Journal of Network Security*, vol. 19, no. 3, pp. 347–357, 2017.
[13] M. Karami, A. Keshavarz, H. Jelodar, "Evaluating the segmentation methods of image logs to identify natural fractures in hydrocarbon wells," *International Journal of Electronics and Information Engineering*, vol. 6, no. 2, pp. 94–109, 2017.
[14] F. Li, Q. Mao, C. C. Chang, "A reversible data hiding scheme based on iwt and the sudoku method," *International Journal of Network Security*, vol. 18, no. 3, pp. 410–419, 2016.
[15] L. Liu, Z. Cao, "Analysis of two confidentiality-preserving image search schemes based on additive homomorphic encryption," *International Journal of Electronics and Information Engineering*, vol. 5, no. 1, pp. 1–5, 2016.
[16] H. C. Lu, Y. P. Chu, M. S. Hwang, "A new steganographic method of the pixel-value differencing", *The Journal of Imaging Science and Technology*, vol. 50, no. 5, pp. 424–426, 2006.
[17] T. C. Lu, C. Y. Tseng, J. H. Wu, "Dual imaging-based reversible hiding technique using LSB matching," *Signal Processing*, vol. 108, pp. 77–89, 2015.
[18] T. C. Lu, J. H. Wu, C. C. Huang, "Dual-image-based reversible data hiding method using center folding strategy," *Signal Processing*, vol. 115, pp. 195–213, 2015.
[19] S. Manoharan, D. RajKumar, "Pixel value differencing method based on CMYK colour model," *International Journal of Electronics and Information Engineering*, vol. 5, no. 1, pp. 37–46, 2016.
[20] A. Mehran, D. M. Chandler, "Digital watermarking via adaptive logo texturization," *IEEE Transactions on Image Processing*, vol. 24, no. 12, pp. 5060–5073, 2015.
[21] M. Shobana, "Efficient X-box mapping in stego-image using four-bit concatenation," *International Journal of Electronics and Information Engineering*, vol. 1, no. 1, pp. 29–33, 2014.
[22] J. Tian, "Reversible data embedding using a difference expansion," *IEEE Transactions on Circuits System Video Technology*, vol. 13, no. 8, pp. 890–896, 2003.
[23] Y. L. Wang, J. J. Shen, M. S. Hwang, "An improved dual image-based reversible hiding technique using LSB matching," *International Journal of Network Security*, vol. 19, no. 5, pp. 858–862, 2017.
[24] Y. L. Wang, J. J. Shen, M. S. Hwang, "A survey of reversible data hiding for VQ-compressed images," *International Journal of Network Security*, vol. 20, no. 1, pp. 1–8, 2018.
[25] N. I Wu and M. S. Hwang, "Data hiding: Current status and key issues", *International Journal of Network Security*, vol. 4, no. 1, pp. 1–9, 2007.
[26] N. I Wu, M. S. Hwang, "Development of a data hiding scheme based on combination theory for lowering the visual noise in binary images," *Displays*, vol. 49, pp. 116–123, 2017.

Biography

Yu-Lun Wang received his B.S. degree in information management from Fu Jen Catholic University in 2014 and M.S. degree in Management Information Systems from National Chung Hsing University in 2016. His research interests include steganography and digital image.

Jau-Ji Shen received his Ph.D. degree from National Taiwan University in 1988. His research interests include digital image, software engineering, information security, and data base technique. His work experiences include the Director of National Formosa University Library and the Associate Dean of Management School in Chaoyang University of Technology. Now, he is a professor in the Department of Management Information Systems, National Chung Hsing University.

Min-Shiang Hwang received the Ph.D. degree in computer and information science from the National Chiao Tung University, Taiwan in 1995. Dr. Hwang was the Chairman of the Department of Information Management, Chaoyang University of Technology (CYUT), Taiwan, during 1999-2003. He was a distinguished professor and Chairman of the Department of Management Information Systems, NCHU, during 2003-2011. He obtained the 1997, 1998, 1999, 2000, and 2001 Excellent Research Award of National Science Council (Taiwan). His current research interests include information security, cryptography, image compression, and mobile computing. Dr. Hwang has published over 200+ articles on the above research fields in international journals.

Defense of Computer Network Viruses Based on Data Mining Technology

Cen Zuo

(Corresponding author: Cen Zuo)

Chongqing College of Electronic Engineering
Shapingba, Chongqing 401331, China
(Email: cenzuocq@126.com)

(Received Dec. 12, 2017; revised and accepted Apr. 6, 2018)

Abstract

Computer network has great influence on people's work and life. Enterprise information, personal information and even national information are stored in computers. Therefore, computer network security has become a problem that cannot be ignored. The biggest threat to computer network security is computer network virus. To cope with the invasion of different viruses, this study analyzed the characteristics of network viruses and designed computer data mining module by combining data mining technology with dynamic behavioral intercept technique to mine hidden information and determine whether there was virus. The method was applied in the detection of Trojan horse virus on network. The efficacy of defense of Trojan horse viruses was tested through indexes including false alarm rate, accuracy rate, omission rate and information gain value extracted based on API characteristics. The detection suggests that data mining technology is useful in the defense of computer network viruses and has a favorable development prospect.

Keywords: Computer Network; Data Mining Technology; Virus

1 Introduction

Computer network viruses spread with the constant development of computer network technology. Computer virus can be created through compilation on advanced program, and other viruses can be derived through modification. Therefore there are diverse network viruses with certain uncertainty [1, 2]. El-Sayed *et al.* [3] established mathematical model for the transmission process of computer viruses, which is beneficial to the understanding of computer virus behaviors and prevention of viruses. They put forward fractional order SIR (Susceptible, infective and removal) model to study computer virus.

To effectively resist computer viruses, Shahrear *et al.* [4] proposed a compartmental model, made changes for each compartment, and analyzed the local stability of virus-free and endemic disease equilibrium models based on basic reproduction number. Trojan horse virus was selected as the research subject in this study. Differing from other viruses which are capable of self-reproduction, Trojan horse viruses will not intentionally infect other documents but confuse users to download and then invade host to steal document information [5]. Feature code scanning technology, active defense technology and network monitoring [6] are the main technologies for the detection of Trojan horse viruses currently. Data mining technology was used in this study. Application of data mining technology in network virus defense system is a new idea for enhancing computer network security [7]. The technology takes data in a certain range as the research subjects and collects, analyzes and classifies them; the processing result is regarded as the determination basis for a potential relationship and data regularity [8]. The preparation of data and searching and presentation of data regularity are the important components of data mining technology [9]. It is found that data mining has a high accuracy, low omission ratio and low false alarm rate in the detection of Trojan horse viruses, suggesting data mining has a bright development prospect in the detection of computer viruses.

2 Data Mining Technology

Three important components of data mining technology can be subdivided into data preprocessing module, decision-making module, data collection module, data mining module and rule base module [10, 11]. The main process steps of data mining technology are data collection, preprocessing, data cleaning, data mining, modeling and model evaluation.

2.1 Classification Algorithm

The common data mining classification algorithms mainly includes Bayesian algorithm, support vector machine and

decision-making tree. Bayesian algorithm is to calculate the prior probability of a known object firstly based on Bayesian classifier and then the probability of a category using theorem formula. Bayesian classifier can optimize classification model based on minimum error rate. The algorithm has advantages of high preciseness and simple operation. Support vector machine can map linearly inseparable samples in a low-dimensional space to linearly separable samples in a high-dimensional space. Moreover it can establish optimal segmentation plane in data space and learn globally optimal solution. Decision-making tree is a tree-structure model composing of branches in data structure. It represents a decision making system and is a frequently used multi-layer classifier in data mining [12].

2.2 Application Programming Interface Function

Data mining technology is extensively applied. Detection of Trojan horse viruses, defense of viruses and network monitoring all need data mining technology. Especially in the detection of Trojan horse viruses, it can distinguish Trojan horse viruses with normal documents through learning relevant rules based on the features of Trojan horse viruses. Application programming interface (API) function and relevant parameters were regarded as the data set. The behavioral characteristics of Trojan horse viruses were tracked, and signal gain was assigned with characteristic value. Finally features were extracted through effective classification algorithm, with high accuracy and small error.

3 Data Mining Technology - Taking Trojan Horse Viruses as an Example

Many API functions need to be called if there are a larger amount of malicious codes. Therefore it is critical to mine combinations which are beneficial to the detection of malicious codes in API function. The structural chart of the detection system is shown in Figure 1.

As shown in Figure 1, the system was mainly composed of tracking module, preprocessing module and data mining and analysis module. The tracking module was responsible for analyzing dynamic link library (DLL) formed in sample operation, comparing it with illegal behaviors captured in training stage, and monitoring specified API function. When the defined API function was detected, the system interrupted, and then the function given by the current parameters was analyzed. Preprocessing module was responsible for loading program to sample database in batches. Mining and analysis module is the decision-making module of the system. API functions which were called by Trojan horse viruses could also be called in normal documents; the combination of those functions threatened the safety of computer network. Then the data mining module was designed in details.

3.1 Tracking and Monitoring of API Function

API function is usually acquired through Windows Debug AP and APIHOOK. API function was the core in the tracking module. Windows Debug API could call API, load a program, and bind itself on running programs; moreover it could collect image bases addresses and call addresses in the process of debugging. If an incident was correlated to debugging, debugger would be immediately triggered, and monitoring process would be activated, including creating process, thread, loading and releasing. Debugger could respond to and handle incidents. Submodule was responsible for the interactive behaviors between the process of debugger and debugged process and help debuggers realize different debugging functions. In short, API function could coordinate with debugging submodules.

3.1.1 Procedures of Tracking and Monitoring

The first step was to input samples, start monitoring program, and suspend all threads. The second step was to analyze library files which were called by debugging programs to acquire derived functions and the entry addresses of different functions. The third step was to set breakpoints on the entry addresses of functions. The fourth step was to debug incidents and collect the information of debugging process if there was interruption at the breakpoints. The fifth step was to execute debugging process repeatedly till the end and acquire behavioral reports.

3.1.2 Implementation of Monitoring

The first step was to obtain the names of process and process modules in EnumProcess function, including enumerating process, enumerating process module, and acquiring the names of process modules.

1) Enumerating process:

```
typedef BOOL(_stdcall*ENUMPROCESSES)(
    DWORD*pProcessIds, \\ ID array chains of
                          \\ pointing process
    DWORD*cb, \\ Size of ID array
    DWORD*pBytesReturened \\ Bytes returned
    );
```

2) Enumerating process module:

```
typedef BOOL(_stdcall*ENUMPROCESSESMODULES)(
    HANDLEh_Process, \\ Process handle
    HMODULE*lphModule, \\ Array chain
    DWORD cb, \\ Size of array (bytes)
    LPDWORD lpcbNeeded \\ Total number of
                  \\ bytes needed by memory handle
```

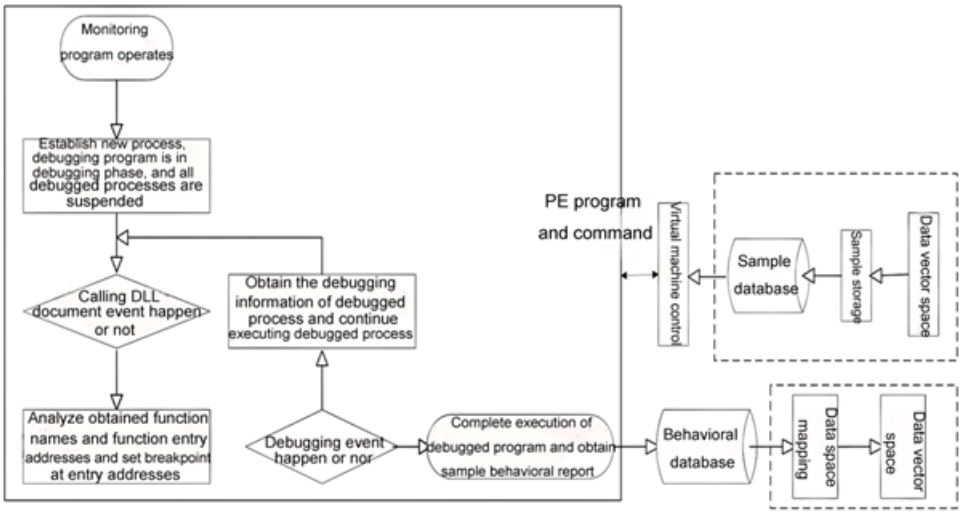


Figure 1: The flow chart of the detection system

);

3) Acquiring the names of process modules;

typedef DWORD(_stdcall*GETMODULEFILENAMEX)(
HANDLE hProcess, \\ Process handle
HMODULE hModule, \\ Process handle
LPTSTR lpFileName, \\ Full path of module storage
DWORD nSize \\ lpFileName Size of buffer (character)
);

3.2 Data Mining and Analysis

WEKA platform is the most commonly used analysis tool in data mining. It is an intelligent analysis platform based on Java environment which is feasible to all operation environments including data preprocessing, cluster analysis and classification prediction.

3.2.1 Vector Space

In the capture of API function by the tracking module, function was regarded as a feature, and sample vector space was established for normal samples and Trojan horse virus samples separately. If the API function was in Trojan horse virus, then the characteristic value was set as 1; if it was not, then the characteristic value was set as 0. In this way, vector space could be established for legal programs, all samples could be represented as the vectors of the vector space, and relevant models could be established using data mining technology. In the construction of data set, ID stands for the number of program file sample, Function stands for whether there was presence of corresponding function in the sample, and Type stands for the type of corresponding sample, normal file or Trojan horse virus.

The model building of characteristic value is as follows. The entropy of random variable E was:

$$H(E) = -\sum_{i=1}^{n} p_i(E = c_i) \log_2 p_i(E = c_i), \tag{1}$$

where there were n values for E and pi stands for the probability when E was equal to ci. Entropy could reflect the distribution condition and distinction degree of E. High value of entropy indicated uniform distribution and low distinction degree, while small value of entropy shows large distinction degree. When random variable G was known, the conditional entropy of E was:

$$H(E|G) = -\sum_{s=1}^{m} p_s(Y = c_s) \times H(E|G = c_s). \tag{2}$$

$H(E|G)$ means E was still uncertain even if G was known. Random variable E could be either malicious code or non-malicious code. Hence

$$H(E) = -\frac{|c|}{|x|} \log_2 \frac{|c|}{|x|} - \frac{|a|}{|x|} \log_2 \frac{|a|}{|x|} \tag{3}$$

where c stands for program containing malicious code, a stands for normal program, $|x| = |c| + |b|$ stands for the total number of programs, and G stands for specified API function. Then

$$H(E|G) = -\sum_{n=0}^{1} \frac{|x_n|}{|x|} (-\frac{|c_n|}{|x_n|} \log_2 \frac{|c_n|}{|x_n|} - \frac{|a_n|}{|x_n|} \log_2 \frac{|a_n|}{|x_n|}) \tag{4}$$

Information gain $IG(E|G)$ was:

$$IG(E|G) = H(E|G) - H(X). \tag{5}$$

The above calculation was based on the difference of probability when specific API function program contained malicious code and probability when program which did not call the API function contained malicious code. A large difference indicated a high probability of the presence of API function in viruses. Through calculation, characteristics which had large effects in the detection were reserved.

3.2.2 Validation Method

Cross validation was involved in this study. K-fold cross validation could randomly divide data set into k subsets. One subset was taken as the test set each time, and the remaining subsets were regarded as the training sets. After k times of repetition, the average testing result was taken as the final result. K-fold complete cross validation was adopted as the division of data had significant fluctuation. That validation method could take all possible division methods that could divide the data set into k subsets into account. K-fold cross validation repeated for n times, and the average value was taken as the final result. K-fold complete cross validation is complete though its accuracy is high. Hence K-fold hierarchical cross validation was proposed. Then the information gain values of Kernel32. dll function and its parameters in the detection of Trojan horse viruses were statistically analyzed.

3.3 Interception Method for Function Call

Interception methods for function call mainly included modifying System Service Descriptor Table (SSDT), monitoring CreateremoteThread, monitoring NtCreateProcessEx function and intercepting NtCreateProcessEx function. The operation of process initiation could be realized by intercepting function through modifying SSDT or shadow SSDT. In the monitoring of CreateremoteThread, CreateremoteThread was the tool for remote injection of Trojan horse virus. Monitoring CreateremoteThread was helpful to the checking of system abnormality. The main aim of monitoring NtCreateProcessEx function and intercepting NtCreateProcessEx function was to intercept through API function, achieving the detection of Trojan horse viruses through process monitoring, and introducing new safety monitoring system. In the process of interception, the path of function call and document names were acquired, as shown in Algorithm 1.

Algorithm 1 The process of interception

```
1: GetModuleFileName(NULL, cur_mod, sizeof(cur_mod));
2: while (pMyAPIInfo[count].module_name != NULL)
   do
3:    Strcpy(pszModuleName, pMyAPIInfo[count].module_name);
4:    Strcpy(pszModuleName, pMyAPIInfo[count].function_name);
5:    Strcpy(pszMyFuncName, pMyAPIInfo[count].myfunc);
6: end while
7: if (pszAPIName!=NULL) then
8:    pszParameterList = strchr(pszAPIName,'('));
9: end if
10: if (pszParameterList != NULL) then
11:    pszParameterList[0] = '\0';
12:    pszParameterList++;
13: end if
14: if ((myFunc=(tagMyFunc)GetProcAddress(hMyDLL,
    pszMyFuncName)) = NULL) then
15:    Return false;
16: end if
17: pAPIInfo=HookAPIfunction(pszModuleName,
    pszAPName, myFunc);
18: ...
```

4 Evaluation Based on Tests

The false alarm rate, accuracy and omission rate were taken as the evaluation indexes for the detection system. False alarm rate referred to the percentage of the normal files which were wrongly classified as Trojan horse viruses; accuracy rate referred to the percentage of program files which were evaluated accurately; omission rate referred to the percentage of Trojan horse viruses which were wrongly evaluated as normal files. In the test, there were 900 legal files and 1300 Trojan horse viruses.

4.1 Native API Related Information After Interception

Table 1 exhibits Native API related information after program interception. Through comparison and analysis, the intercepted API function was regarded as a characteristic for establishing data format of the whole WEKA (Waikato Environment for Knowledge Analysis) analysis. The data were analyzed using Naive Bayesian algorithm, support vector machine and J48 algorithm.

4.2 Analysis of Trojan Horse Virus Identification with Three Algorithms

Table 2 exhibits that the three algorithms performed differently in identifying Trojan horse viruses. In conclusion, defense based on data mining had favorable effect in identifying Trojan horse viruses; all the three algorithms could identify more than 1200 Trojan horse viruses (90%).

4.3 K-fold Hierarchical Cross-validation

The average results of the false alarm rate, accuracy and omission rate when K-fold hierarchical cross-validation was sued are shown in Table 3.

Table 3 exhibits that the detection method based on data mining performed best in the detection of Trojan horse viruses because of its high detection rate and low false alarm rate. Among the three algorithms, the omission rate of Naive Bayesian algorithm was significantly higher than that of the other two algorithms; the accuracy of support vector machine was the highest; the false alarm rate of J48 algorithm was the highest, and next was Naive Bayesian algorithm. Overall the accuracy of them was all high, larger than 95%. They were able to make beneficial determination based on the vector space constructed by intercepted API function, and moreover the technology was verified as quite effective in detecting Trojan horse viruses.

Table 1: Native API related information after interception

Timestamp	API ID	Native API
17	4	NtAllocate VirtualMemory (allocate virtual memory)
18	23	NtQueryVirtualMemory (query virtual memory)
19	15	NtFreeVirtualMemory (free virtual memory)
20	7	NtSetEvent (setting event object)
21	16	NtCreateEvent (creating event object)
22	18	NtCancelTimer (cancel current time setting)
23	34	NtSetTimer (resetting time)
24	69	NtDelayExecution (delay execution)
25	44	NtClearEvent (clearing event object)
26	9	NtOpenThreadToken (opening thread mark)
27	8	NtAllocateVirtualMemory (allocate virtual memory)

Table 2: Relevant information for identification of Trojan horse viruses

	Number of Trojan horse viruses identified	Number of Trojan horse viruses identified as legal	Number of non-classified
Naive Bayesian algorithm	1223	71	6
Support vector machine	1270	28	2
J48 algorithm	1265	32	3

4.4 Analysis of Kernel32.dll Parameters in the Detection

Table 4 suggests the information gain values of Kernel32.dll function in the detection of Trojan horse viruses. Features were extracted through feature extraction algorithm, and the file features which existed or did not exist were examined. The values of information gain were small, indicating that the reserved function had high distinction degree and the classification efficiency was high.

5 Discussion and Conclusion

Computer plays an increasingly larger role in the life and work of people in the process of its continuous development and improvement, which imposes a huge threat to the security of computer network. There is no effective way to prevent viruses from intruding into computer network, and only defense is feasible [13]. Defense of network viruses is challenging as computer network viruses have multiple transmission modes and strong pertinence. Trojan horse virus as a kind of computer network viruses is occult; hence monitoring Trojan horse viruses simply and blindly will consume a large number of system resources, leading to the occurrence of false alarm [14, 15]. Data mining can excavate processes concealed, analyze API function call, summarize a new detection scheme, and establish new detection system to reduce false alarm rate. In this study, API tracking module was used, and the false alarm rate, accuracy and omission rate of three algorithms were tested to investigate the role of data mining technology in the detection of Trojan horse viruses. Moreover it was found from the extraction of information gain based on the characteristic values of API function that the classification efficiency was improved. Data mining could make beneficial judgment through the vector space established based on the intercepted API function, which enhanced identification efficiency. Hence data mining is quite effective in detecting Trojan horse viruses. In conclusion, data mining technology is useful in the detection of computer network viruses, which is worth promotion.

References

[1] C. H. Zhai, "Discussion on the utilization of prevention Technology of Computer Network Security," *Applied Mechanics and Materials*, vol. 556-562, pp. 5523–5525, 2014.

[2] R. K. Upadhyay, S. Kumari, A. K. Misra, "Modeling the virus dynamics in computer network with SVEIR model and nonlinear incident rate," *Journal of Applied Mathematics and Computing*, vol. 54, no. 1-2, pp. 485–509, 2017.

[3] A. M. A. El-Sayed, A. A. M. Arafa, M. Khali, A. Hassan, "A mathematical model with memory for propagation of computer virus under human intervention," *Progress in Fractional Differentiation and Applications*, vol. 2, pp. 105–113, 2016.

[4] P. Shahrear, A. K. Chakraborty, M. A. Islam, U. Habiba, "Analysis of computer virus propagation based on compartmental model," *Applied and Computational Mathematics*, vol. 7, no. 1-2, pp. 12–21, 2018.

[5] M. M. Saudi, A. M. Abuzaid, B. M. Taib, Z. H. Abdullah, "Designing a new model for Trojan horse de-

Table 3: The average test results

	False alarm rate	Accuracy	Omission rate
Naive Bayesian algorithm	2.4%	95.61%	5.5%
Support vector machine	1.6%	98.08%	2.2%
J48 algorithm	3.1%	97.26%	2.5%

Table 4: The information gain values of Kernel32.dll parameters in the detection of Trojan horse viruses

Kernel32.dll	Parameters	IG	Description of functions
Open process	dwDesiredAccess	12.12%	Needed process rights
	bInheritHandle	6.02%	Whether can inherit or not
Allocation space	hProcess	14.23%	Process handle
	IpAddress	4.56%	Address pointer
	flProtect	4.56%	Page protection attribute
	dwSize	4.56%	Size

tection using sequential minimal optimization," *Lecture Notes in Electrical Engineering*, vol. 315, pp. 739–746, 2015.

[6] J. A. Ortega, D. Fuentes, J. A. Alvarez, L. Gonzalez-abril, F. Velasco, "A novel approach to Trojan horse detection in mobile phones messaging and Bluetooth services," *KSII Transactions on Internet and Information Systems*, vol. 5, no. 8, pp. 1457–1471, 2011.

[7] L. P. Feng, H. B. Wang, S. Q. Feng, "Computer network virus propagation model based on biology principle," *Computer Engineering*, vol. 37, pp. 155–157, 2011.

[8] H. T. Tavani, "Genomic research and data-mining technology: Implications for personal privacy and informed consent," *Ethics and Information Technology*, vol. 6, no. 1, pp. 15–28, 2004.

[9] J. J. Xie, "Integration of GIS and data mining technology to enhance the pavement management decision making," *Journal of Transportation Engineering*, vol. 136, no. 4, pp. 332–341, 2010.

[10] Q. Jiang, H. Lin, J. Li, J. Liu, "The research on spatial data mining module based on multi-objective optimization model for decision support system," in *IEEE Second WRI Global Congress on Intelligent Systems (GCIS'10)*, pp. 299–302, 2010.

[11] C. Chen, A. Chen, "Using data mining technology to provide a recommendation service in the digital library," *The Electronic Library*, vol. 25, no. 6, pp. 711–724, 2007.

[12] P. Sangitab, S. R. Deshmukh, "Use of support vector machine, decision tree and naive Bayesian techniques for wind speed classification," in *International Conference on Power and Energy Systems (ICPS'11)*, pp. 1–8, 2011.

[13] P. Kata, *The in Vitro Effects of Herpes Simplex Virus and Rubella Virus on Autophagy*, Ph.D. Thesis, University of Szeged, 2014.

[14] B. Liu, C. Lin, Q. Jian, J. He, P. Ungsunan, "A NetFlow based flow analysis and monitoring system in enterprise networks," *Computer Networks*, vol. 52, no. 5, pp. 1074–1092, 2008.

[15] S. Zhong, X. Cheng, T. Chen, "Data hiding in a kind of PDF texts for secret communication," *International Journal of Network Security*, vol. 4, pp. 17–26, 2007.

Biography

Cen Zuo is a teacher from the school of computing of Chongqing College Of Electronic Engineering, China. His interests of research are computer application and software engineering. He gained the bachelor's degree from the software college of Chongqing University of Posts and Telecommunications in 2009 and now is studying for a master's degree in the Computer College of Chongqing University. He have participated in many provincial-level subjects such as Research and Practice of Fusion of Innovative and Entrepreneurial Education of Higher Vocational College and Status Analysis and Countermeasure Study of Modern Apprenticeship Practice in Higher Vocational College and published more than 10 papers including one EI indexed paper titled Research on Scheduling Algorithm for Cloud Computing and one CSCD core paper titled The Route Optimization of the Shortest Path with Fruit Fly Optimization Algorithm, and co-edited a computer science textbook titled Management Information System Design Practice Guide. Moreover he taught courses such as Fundamentals of Computer Culture, Program Design Fundamentals and Database Fundamentals. He also gained honorary titles such as school-level excellent education workers.

Guide for Authors
International Journal of Network Security

IJNS will be committed to the timely publication of very high-quality, peer-reviewed, original papers that advance the state-of-the art and applications of network security. Topics will include, but not be limited to, the following: Biometric Security, Communications and Networks Security, Cryptography, Database Security, Electronic Commerce Security, Multimedia Security, System Security, etc.

1. Submission Procedure

Authors are strongly encouraged to submit their papers electronically by using online manuscript submission at http://ijns.jalaxy.com.tw/.

2. General

Articles must be written in good English. Submission of an article implies that the work described has not been published previously, that it is not under consideration for publication elsewhere. It will not be published elsewhere in the same form, in English or in any other language, without the written consent of the Publisher.

2.1 Length Limitation:

All papers should be concisely written and be no longer than 30 double-spaced pages (12-point font, approximately 26 lines/page) including figures.

2.2 Title page

The title page should contain the article title, author(s) names and affiliations, address, an abstract not exceeding 100 words, and a list of three to five keywords.

2.3 Corresponding author

Clearly indicate who is willing to handle correspondence at all stages of refereeing and publication. Ensure that telephone and fax numbers (with country and area code) are provided in addition to the e-mail address and the complete postal address.

2.4 References

References should be listed alphabetically, in the same way as follows:
For a paper in a journal: M. S. Hwang, C. C. Chang, and K. F. Hwang, ``An ElGamal-like cryptosystem for enciphering large messages," *IEEE Transactions on Knowledge and Data Engineering*, vol. 14, no. 2, pp. 445--446, 2002.

For a book: Dorothy E. R. Denning, *Cryptography and Data Security*. Massachusetts: Addison-Wesley, 1982.

For a paper in a proceeding: M. S. Hwang, C. C. Lee, and Y. L. Tang, ``Two simple batch verifying multiple digital signatures," in *The Third International Conference on Information and Communication Security (ICICS2001)*, pp. 13--16, Xian, China, 2001.

In text, references should be indicated by [number].

Subscription Information

Individual subscriptions to IJNS are available at the annual rate of US$ 200.00 or NT 7,000 (Taiwan). The rate is US$1000.00 or NT 30,000 (Taiwan) for institutional subscriptions. Price includes surface postage, packing and handling charges worldwide. Please make your payment payable to "Jalaxy Technique Co., LTD." For detailed information, please refer to http://ijns.jalaxy.com.tw or Email to ijns.publishing@gmail.com.